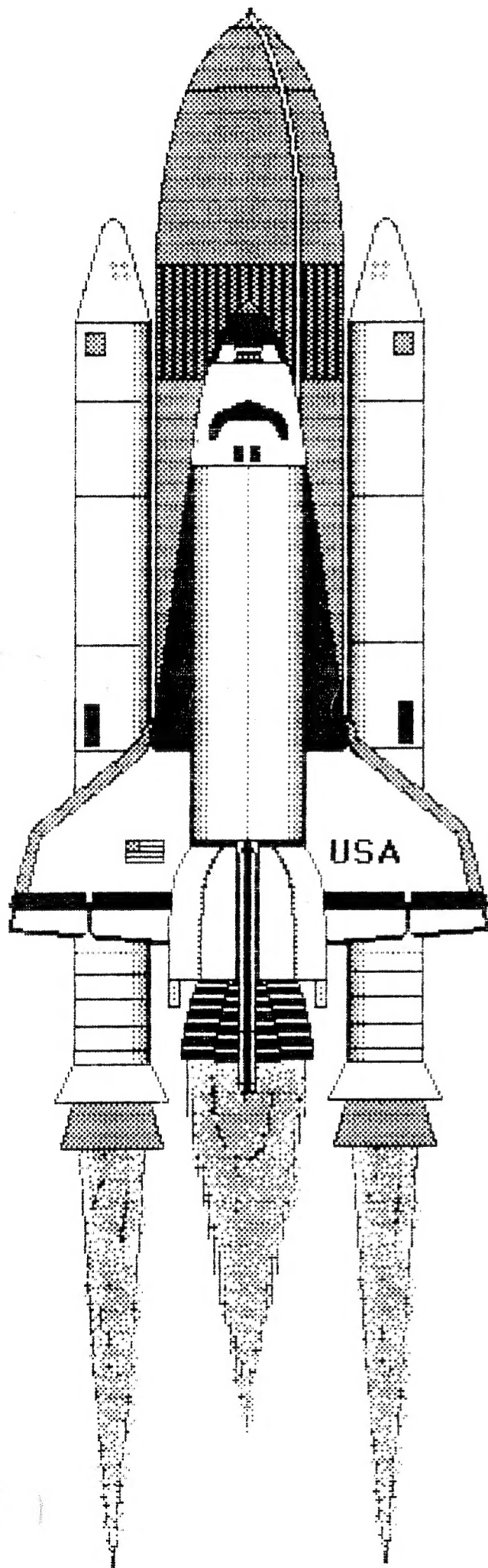


SINC - LINK

JUL-AUG '91
VOL. 9 NO. 4



LATE SUMMER ISSUE

page 2	Club Stuff
page 3	Editorial
page 4,5	Bob's Notebook - Toolkit Part 1 (2068)
page 6	QLips (QL)
page 7	Timex Logic - Logic Operator NOT (ZX81,2068)
page 8,9,10	Sort of Sorts (2068)
page 11	Byte Power Ad (2068)
page 12	Minerva and Others (QL)
page 13	Fixing My FDD3000 Disk System (2068)
page 14	QL Help Request (QL)
page 15	Letter to Secretary re Vender
page 16	Advertisements
page 17	Quick Directory Sorter (2068)
page 18,19	Bob's Notebook - Toolkit Part 2 (2068)
page 20,21	Hex Loader Circle Program (2068)
page 22	MDV Problem and Other Requests (All)
page 23	Do We Need A Central Organization? (All)
page 24	Bob's Notebook Correction (2068)
page 25,26	ZX81 - "PRINT USING" (ZX81/TS1000)
page 27,28	QL Primer (QL)
page 29	More Tasword II (2068)
page 30	Mastering "MERGE" (2068)
page 31	Disk Menu Loader (2068)
page 32	Mike's Notebook - Disk CAT Loader (2068)
page 33,34	Software Review - "VIDEOTEX v1.5" (2068)

SINC - LINK

SINC-LINK IS A PUBLICATION OF THE TORONTO TIMEX-SINCLAIR USERS CLUB AND IS ISSUED 6 TIMES A YEAR. COPIES OF THE NEWSLETTER ARE \$1.50 EACH FOR NON-MEMBERS. CLUB MEMBERS RECEIVE FREE COPIES AS PART OF THE \$20.00 ANNUAL MEMBERSHIP FEE. A NEWSLETTER SUBSCRIPTION ONLY IS AVAILABLE FOR \$12.00.

NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USERS GROUPS.

PLEASE CREDIT THIS PUBLICATION AND THE AUTHOR IF YOU COPY MATERIAL.

THE CLUB MEETS ON THE FIRST WEDNESDAY OF EACH MONTH AT FOREST HILLS COLLEGIATE INSTITUTE, 730 EGLINTON AVE. W., TORONTO.

SINC-LINK IS PRODUCED ENTIRELY ON SINCLAIR AND TIMEX-SINCLAIR COMPUTERS.

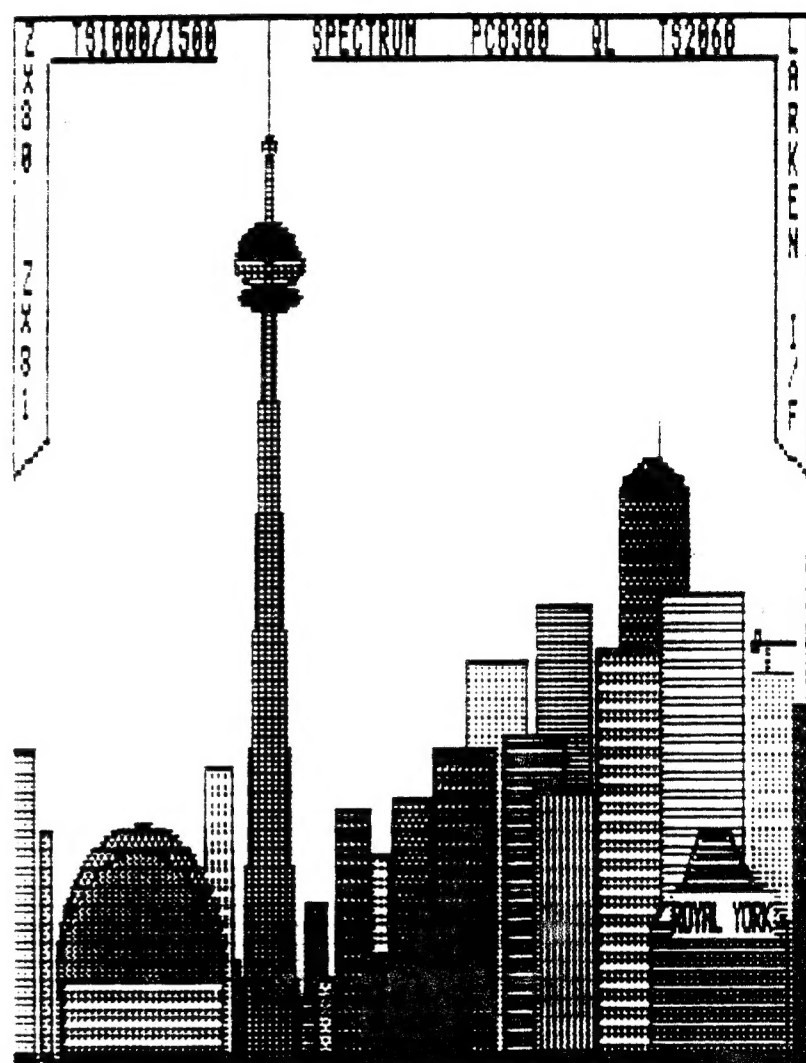
SEND CORRESPONDANCE TO:

Attention: SINC-LINK EDITOR
TORONTO TIMEX-SINCLAIR USERS
CLUB, 14 RICHOME COURT,
SCARBOROUGH, ONTARIO,
CANADA M1K 2Y1

EXECUTIVE OFFICERS:

PRESIDENT:	RENE BRUNEAU (531-9749)
TREASURER:	BILL LAWSON (444-8772)
SECRETARY:	GEORGE CHAMBERS (751-7559)
ACTIVITIES:	RENE BRUNEAU (531-9749)
TAPE LIBRARIAN ZX81:	LYMAN PAQUETTE (482-4479)
TAPE/DISK LIBRARIAN QL:	HUGH HOWIE (634-4929)
PAPER LIBRARIAN:	JEFF TAYLOR (244-8583)
NEWSLETTER:	JEFF TAYLOR (244-8583)
LIAISON OFFICER:	GEORGE CHAMBERS, 14 RICHOME COURT,
(Out-of-town members)	SCARBOROUGH, ONTARIO, M1K 2Y1
	(416-751-7559)

JUL-AUG '91
VOL. 9 NO. 4



TORONTO TIMEX-SINCLAIR
USERS CLUB

TORONTO TIMEX-SINCLAIR USERS CLUB

Editorial

One of the problems with producing this newsletter is that my material comes from many sources and many printers. While I'm very happy to receive all this good stuff from our numerous contributors, I'm often less than thrilled with the print quality of the copy I get. Read that carefully: I'm happy with content, not always happy with product.

I try to make Sinc-Link look as good as possible but some of the articles I get are so faded or lightly printed that even with the considerable print-enhancing features of my copier I can still only produce a barely readable copy. Since I promise to print everything I get, and it's all excellent material, this sometimes hurts how the newsletter looks. What to do?

I could retype and print all the articles myself but I have neither the time nor the inclination to do so. You could send your articles on tape or disk and I'd be happy to print them out but it might not look as good as your version and the newsletter would acquire a sort of boring sameness. What to do?

There is a simpler solution, of course. Keep your printer ribbon fresh. Either replace it or re-ink it regularly. What's the point of going to the trouble of writing an article for people to read if people can't read or copy the end product? Think about it, please.

To this end I've included a simple program for TS2068 users of Epson-compatible printers and the Larken system. Simply install it, run it, then load your word processor. When you print in NLQ mode your characters will be struck four times. Makes text nice and dark. Enough whining, already.

```
1 REM Renato Zannese, Toronto Timex-Sinclair Users Club
2 REM Print Enhancer
5 RANDOMIZE USR 100: OPEN #3,"lp"
10 RANDOMIZE USR 100: POKE 16093,32
20 LPRINT CHR$ 9;CHR$ 27;CHR$ 104;CHR$ 0
21 REM Change CHR$ 0 to 1 for double-height text or to 2 for
quad-height text.
25 LPRINT CHR$ 27;CHR$ 71;CHR$ 9
30 RANDOMIZE USR 100: POKE 16093,0
40 LPRINT "This is Normal-Sized, Double-Struck."
```

Bye, Renato

Renato Zannese, long-time club member, newsletter contributor and club 2068 tape librarian has moved on to bigger (and better?) things. This is the guy who first introduced me to this club back in my TS1000 days and found me my first TS2068. Several members have his own-designed Kempston interface and have had defective computers fixed by him. The program above is one of his little gems. Bye, Renato, and good luck.

Cover

The Space Shuttle and text are from Byte Power's "Print Factory" and "Companion" desk top publisher and super screen suite for the 2068. To date, I've only seen one other publication, Mike Felerski's excellent "Sinclair Desktop Publishing Journal", utilizing this great package of programs. Is anyone else out there? I'd like to see some samples. Write, please.

That's all for now...

J.T.



BOB'S Notebook

by:
Bob Mitchell
20 Uild Briarway
WILLOUDALE Ontario
M2J 2L2

Apr 91

TOOLKIT: Part 1==>RENUMBER FUNCTIONS.

This is the first of a series of tutorials designed to give TS2068 users some guidance on using the various TOOLKIT options. This article will focus on the renumbering functions: RENUMBERING, MOVING and COPYING BLOCKS of LINES. Do NOT use any of the renumber functions with programs that have machine code in REM lines.

BACKGROUND...

The program tstk.B1 with its code tstk.C1 is available from the club library and has also been included in several versions of the Omnibus disk. It originally appeared in the UK magazine Your Computer May 1985 page 91 written by D. Spagnol for the Spectrum computer. I subsequently adapted it to the TS2068 which included a redesign of the interrupt mode machine code section; then I adapted it back again to the Spectrum so that it would work with a Spectrum emulator.

Whenever you are going to use TOOLKIT, LOAD it, ensure both RANDOMIZE USR 59696 AND RANDOMIZE USR 60000 are used to activate it (the former activates the interrupt mode code; the latter brings up the TOOLKIT menu). BREAK back into BASIC using CAPS SHIFT 6 and put in the usual line 9999 RANDOMIZE USR 60000. This line is a MUST if you are going to use Renumber, Move, Copy, Machine Code to Data, UDGs to Data or Create a REM Line. It will stop new lines being created from overflowing into the variables area.

LINE RENUMBERING...

When any of the toolkit line renumbering routines are used, all affected GO TOs, GO SUBs, RESTORES, LINES, even LLISTs and RUNs are corrected throughout the entire listing (providing these all use real numbers and not variables or computed line numbers). Real numbers mean purely integer ones up to four figures in length.

If a computed line number is encountered, such as <GO TO 3190 +(CODE n\$-96)*oj>, TOOLKIT will not affect it. If it finds a variable, eg, GO SUB ml, it will not affect it either, but you must EDIT any line in which such variables are declared and alter the affected values. Make sure the changed variables get altered in the variables area. To do this, you would usually GO TO the program start line. You could of course EDIT the line again and DELETE the line number which will ensure all the changed variables in that line are placed in the variables area.

WARNING ABOUT SAYING...

As you use the various options, it is very wise to SAVE your work many times as you proceed. Something may go wrong. But a word of warning! Do not try to make a LARKEN AUTOSTART SAVE of your new listing unless the interrupts are NORMAL. Check toolkit option I; if the interrupts are diverted, press the X option on the menu to make them normal. Then go ahead with the AUTOSTART SAVE.

MOVE LINES

Let's investigate the MOVE option first. From the toolkit menu, select Move Lines <m> which prompts for start line number and 1st unaffected line number. Enter these and the destination line when prompted followed by the increment (usually 10 but not more than 100). The final prompt is Execute? and if all looks correct, press <Y>. The task will be completed in seconds.

Subroutines (using GO SUB) should always be near the start of any listing (for maximum speed) so these are good candidates for MOVE. Choose a listing with such routines near the end of the listing or all over the place and try moving them up front.

RENUMBER

Using RENUMBER is quite similar but it will ensure that line numbers do not overlap whereas MOVE will check the number of used lines at the destination.

Make sure that there is enough memory left. It is possible to get an Out of Memory report if new line numbers after GO TOs are longer (ie, more digits) than the ones being replaced. If there is insufficient room to fit them in, the program will be corrupted usually beyond redemption. Keep SAVING as you go along, just in case! This warning applies to Move and Copy, too.

COPY LINES

Copying blocks of lines is also very much the same. No GO TO destinations in lines outside the new block are affected but the new block is renumbered within itself. Any GO TOs pointing to an area before the original start of the copied block may be renumbered incorrectly. To prevent this happening, MOVE the block you want to copy to the head of the program, COPY it and MOVE it back to its original position.

ERROR REPORTS...

Error Message

May occur in; (shown in **INVERSE**):

Range Error
Numbers Reversed
Invalid Line Number
Zero Not Allowed
Lines Would Overlap
No Room at Destination
Zero Block!
Task Completed

Increment > 100 **BCMNRTU.**
Start > Finish line **ABCDJKLMPRS.**
Line > 9996 **D> non-existent line.**
self-evident **CMNRTU.**
Recheck what you are attempting **R.**
Not enough unused line numbers **BCEM.**
No actual lines from start to end **CD.**
Done.

TS 2068 TOOLKIT

Alter Program	N	Autoline On
Bytes to DATA	O	Locate Token
C Copy Lines	P	Compactor
D Delete lines	Q	Display Memory
E REM Create	R	Renumber
F REM Delete	S	Search & List
G UDG Designer	T	Trace on
H Hex & Dec	U	UDGs to DATA
I Information	V	List Variables
J Merge Lines	W	Disable NEU
K Upper Case	X	N/O/T/W off
L Lower Case	Y	Uncorrupt
M Move Lines	Z	Line Sort

PRESS A KEY FOLLOWED BY ENTER,
or just ENTER anytime for menu

Q L I P S

By Hugh H Howie

A few months ago I asked if anyone would be interested in the use of a Tutorial in SuperBasic. More importantly I asked if anyone would be willing to provide this service, and act as a Tutor.

Unfortunately I only received one letter in reply. I had a couple of comments made to me directly saying this would be a good thing. No one wrote or expressed a willingness to provide this service. So I let it die there, passing a few comments in the newsletter to the effect that no one was interested in providing this service, and that only one had written on its merits. There the subject was laid to rest. Or so I thought.

There was one person who went to work very quietly to give a tutorial. Howard Clase. His tutorial has appeared as part of a series, of which more is coming.

He chose as his subject a short program by Butch Weinberg, which appeared in the Sept/Oct 1990 issue showing how to print a Directory to printer.

Howard Clase developed the idea into a larger program, explaining as he went along exactly what he was doing, and how to achieve a given end. The first part appeared in the Jan/Feb 1991 issue of Sinc-Link. The second part was in the May/June 1991 issue.

So much attention had been paid to the first part, when the second appeared I was deluged with the query, "Where is the first part of this series? AH well---

There is still more to come. Yes a third installment. It may be in the same issue as this letter, It may not, as Howard has a lot on his plate at the moment, but the third installment will come in due course. He has promised me this.

Now here I was sitting thinking that no one was interested in providing this service, and this guy 'way up in the cold of Newfoundland is keeping his fingers warm typing away on his old QL. Giving us a tutorial, without a word being said, or a great blasting fan-fare of trumpets to announce the Great Coming Attraction. How do you like that?

Well I liked it fine, and I enjoy reading

what he has to say. How many of you have tried the program he is writing? Have you tried it? Did you realise it was a tutorial? Or did you just sit and say "This is something else away beyond me" and lay it aside?

O.K..... The Tutorial so many would like, is being provided right now, and it is in Sinc-Link, the progressive one from the North where the wind blows free and freezes the fingers.

Recently I had a letter from Mr Clase asking me if there had been any feedback on the tutorial. I had to say sorry--

Now here is where I ask all you folks out there to limber up those typing fingers and write and let us know if you like what he, Howard has provided. Write to me and I will pass the news on. Write to Howard and he will pass the message on to me. Write to the Club and I will get the message anyway. The main thing is for you to write. Tell us how you like the series. Tell us how you like the presentation. Tell us what else you would like Howard, (or anyone else for that matter) to provide. Write and tell Howard what you would like him to tackle next. Say thanks for the trouble he is taking.

If you listen to all the weather reports up here, the bad weather is coming up the Miss Valley or from Texas. When I speak to someone from Mass or NY all they say is that all the bad weather comes from Can. Sometimes you just can't win. Till now.

Here we have something GOOD coming from Canada, all the way from 'way up North where the cold freezes the watchamacallits and well damn it all why don't you write someone a note and say what you think. After all, one of the reasons for having a computer is for the Word Processor. So lets use the darn thing.

H J Clase. Box 9947. Station B
Newfoundland. Canada. A1A 4L4.

H H Howie. 586 Oneida Dr. Burlington. Ont.
Canada. L7T 3V3.

The Secretary, on front page of this letter. PLEASE WRITE-----TODAY. §

TIMEX LOGIC The Logic Operator NOT

For many of us, computer logic is a bit difficult to understand, and articles like the one reputed to be by Sharon Z. Aker in the July '88 issue of *UPDATE* muddies the water more. On page 14, under the title of Priority, the article states that NOT B<C is interpreted as (NOT B)>C. This is wrong. NOT applies to the entire expression B<C as the < operator has a higher priority than NOT and the computer will evaluate B<C first. Look at page 228 of the manual for a priority listing.

NOT applied to a condition can result only in a logic value of 0 or 1, regardless of the appearance of the condition. For example, consider.....

NOT X = 50

This is interpreted by the computer as....

NOT (X = 50)

If X does equal 50, the condition, X = 50, is TRUE and results in a logic value of 1 for this condition. Then NOT 1 is a logic value of 0.

Conversely, if X has a value of anything other than 50, the condition is considered false, and X = 50 results in a logic value of 0. Then NOT 0, in turn, has a logic value of 1, and the THEN action will take place.

We can show this by a short test program.....

```
10 LET X = 50
20 IF NOT X = 50 THEN PRINT "Yes"
30 IF NOT (X = 50) THEN PRINT "OK"
40 PRINT "End of test"
```

Now change the value of X in Line 10 to anything but 50 and lines 20 and 30 will print out their strings. Why?

Because X = 50 is not true and it's logic value becomes 0. Now, NOT 0 gives a logic value of 1. As the entire expression has a logic value of 1, the THEN action takes place.

The parentheses in line 30 are not necessary, but if you wish to make the evaluation of an expression clear, use them. The computer simply ignores superfluous parentheses, and evaluates X = 50 first because of the priority of the = operator.

Let's go a step further. The computer considers all logic on a numerical

basis. To the computer, every logic term that is true is assigned the number 1; otherwise it gets a 0. We can put this to a test by adding lines such as the following to our test program.....

```
15 PRINT X = 50
25 PRINT NOT X = 50
35 PRINT NOT (X = 50)
```

and we will get nothing but the numbers 1 and/or 0, on printout of these lines.

You can get the same response by evaluating a logic statement yourself and substituting the resulting logical value in the computer line. Instead of line 20 as written above, write it as.....

20 IF 1 THEN PRINT "Yes"

and the computer will print the string. Change the 1 to 0 and the computer will not print the string. As we said before, the computer recognises everything but 0 as the number 1. So now use something like -3 in place of the 1. Again, the line will print out the string word.

If you get this concept of logic well established in your mind, you will have little or no trouble with the logic NOT from here on in.

There is yet another way of treating logic NOT. You can change any expression wherein it appears to an equivalent one by recalling that it merely reverses logic TRUE and FALSE. In doing so, NOT drops out of the expression. So NOT X = 50 can be replaced by X <> 50. Shall we try another? OK. NOT Y > 10 can be replaced by Y <= 10. I personally don't prefer this way of handling NOT because we tend to forget how the computer itself treats this unique operator.

Warren Fricke
8-5-88

323 1/2 N. Church Street
Bowling Green, OH 43402
May 21, 1990

Dear George,

It seems to me like last time, you asked me something about sorting a database. Something about several people were trying to write a database to take advantage of the RAMDISK, with records of about 64 or 256 bytes.

In answer to the first implied question, for something of that type, if you are actually talking about moving things when you sort them - as opposed to moving pointers - the fastest sort is something called a binary insertion sort. Well, even if you aren't talking about moving things.

Let me back up. There are 3 basic types of sort routines. There is the ever-popular bubble sort, there is the swap sort (which really isn't much different), and there is the insertion sort. There are actually several types of insertion sort as well, one of which is the binary insertion sort.

The reason that a binary insertion sort is better on average than the others is that it requires fewer comparisons on average. That is, given a randomly ordered list of, say 100 items, a typical bubble sort, swap sort, or a standard insertion sort will have to perform $50 \times 99 / 2 = 2475$ comparisons, while a binary insertion sort only performs 580 comparisons.

There is a down side, though. The 2475 is an average figure - it could actually be anywhere from 99 to 4950, where as the binary insertion sort will always perform 580 comparisons. Which means that if the list was almost ordered, it is actually slower, but not by too much. It also involves a little more overhead in determining which numbers it has to check. Which means it may actually take longer for each comparison. But it really does make up for it in the number of comparisons.

Let me pull out a rather extreme example. Suppose you create a Tasword file which is filled with random numbers. I did just this earlier, simply by typing `FOR i=33280 TO 49151: POKE i,INT (10*RND)+48:NEXT i`. This will take about 10 minutes to create a file of 248 lines of random numbers. Now, take your program for sorting Tasword files, and see how long it takes to sort that one. I just finished writing a binary insertion sort for Tasword, and it took me 6.57 seconds to sort this file (I had the computer time it, by entering `PRINT #4: POKE 23672,0: RANDOMIZE USR sort: PRINT (PEEK 23672+256*PEEK 23673)/60`).

Anyway, that answers part of your question. The obvious second question, though, is how to sort something which might occupy a couple of different banks of the RAMDISK. This could be a bit of a problem, since it can't all be kept in main memory at once to sort it. That means it will have to be sorted a bit at a time.

Let's consider an extreme case. Suppose someone has taken up all 8 banks of his RAMDISK. If he wanted to re-sort it, he would have to sort each one separately, and then divide each bank into 8 pieces (since he had 8 banks), and put all the lowest pieces in 1 bank, etc., and sort each bank again.

On second thought, let's forget that. It is way too much hassle. If you are considering files that are too big for main memory, you better forget actually moving everything around. Use an indexed approach - it will be faster and less work anyway. Even if each record is only 64 bytes, this means there will be at most 4096 records in the entire RAMDISK. Each pointer would then need to be 2 bytes, so your index would be 8192 bytes long. You probably also need a list of unused records, just like the list on a disk. You might even consider this like a disk in that sense, just a disk whose blocks are a lot smaller than you are used to.

In some sense, it might be better to have records of 128 bytes or bigger. Then you will have 256 or less records in any given bank, so the pointer could consist of 1 byte for the bank, and 1 for which record in the bank.

Regardless, you will find it much easier to sort an index than to sort the file itself. Instead of having to move all the records, possibly even having to swap them between banks, you only have to move the index.

Just for reference, consider the Tasword file we used earlier. Of that 6.57 seconds it took to sort it, I know that only .3 was actually required for the comparisons and overhead. The other 6.27 seconds was involved in actually moving text. If I could have used an index in that example, the moving pointers would have taken less time than the comparisons. In other words, using an index would probably have accomplished the same effect in under 1 second! In practical terms, it is the difference between sorting your hypothetical database in half a minute and in half an hour.

I am afraid that the same isn't true in BASIC - I have the equivalent programs in BASIC, and there is no real gain in using indexes then. Still, here is what the equivalent program would look like in BASIC:

```

1 REM Indexed Binary sort
2 REM Variables:
3 REM   a$(n)=array to sort
4 REM   i(n) = index
5 REM   n = number of records
6 REM   hi = top of search
7 REM   lo = bottom of search
8 REM   pt = middle of search
9
10 LET i(1)=1
11 FOR i=2 TO n
12 LET t$a$(i)
13 LET hi=i
14 LET lo=0
15 LET pt=int ((hi+lo)/2)
16 IF pt=lo THEN GO TO 25
17 IF a$(i(pt))>t$ THEN GO TO
20
18 LET lo=pt
19 GO TO 15
20 LET hi=pt
21 GO TO 15
25 FOR j=1 TO hi+1 STEP -1
26 LET i(j)=i(j-1)
27 NEXT j
28 LET i(hi)=1
29 NEXT i

```

Sort for Tasword only, using binary insertion routine

```
2 CLEAR 32998
3 LET sort=33000
10 LET a=10: LET b=11: LET c=12: LET d=13: LET e=14: LET f=15
11 LET x=sort-1
12 READ a$: IF a$="END" THEN GO TO 15
13 FOR i=1 TO LEN a$ STEP 3: POKE x,16*VAL a$(i)+VAL a$(i+1): LET x=x+
1: NEXT i
14 GO TO 12
15 PRINT "To use, LOAD file, then:"
16 PRINT '"RANDOMIZE USR ";sort
20 PRINT AT 5,0;"Sort on a different column","(1 <= col <= 64) by:"
21 PRINT '"POKE ";sort-1;","(col)"
999 STOP
1000 DATA "01": REM Default column
1001 DATA "21 00 CD 3E 20 2B BE 28"
1002 DATA "FC 01 40 00 09 7D E6 C0"
1003 DATA "6F E5 21 00 82 01 40 00"
1004 DATA "09 D1 ED 52 D0 19 D5 11"
1005 DATA "C0 81 ED 53 7A 81 E5 E5"
1006 DATA "ED B0 E1 22 7C 81 2A 7C"
1007 DATA "81 ED 5B 7A 81 A7 ED 52"
1008 DATA "CB 1C CB 1D 7D E6 C0 6F"
1009 DATA "B4 28 0D 19 E5 CD 5E 81"
1010 DATA "E1 38 E0 22 7A 81 18 DE"
1011 DATA "E1 ED 5B 7C 81 A7 E5 2B"
1012 DATA "ED 52 38 16 44 4D 19 03"
1013 DATA "11 40 00 EB 19 EB ED B8"
1014 DATA "23 11 C0 81 01 40 00 EB"
1015 DATA "ED B0 E1 18 A0 00"
1019 REM Compare subroutine
1020 DATA "3A E7 80 3D 47 3E 40 11"
1021 DATA "C0 81 28 06 90 C8 23 13"
1022 DATA "10 FC 47 1A BE C0 23 13"
1023 DATA "10 F9 C9 00"
1030 DATA "END"
9800 RANDOMIZE USR 100: SAVE "Tasort.CI"CODE 32999,146
9810 STOP
9900 RANDOMIZE USR 100: SAVE "Tasort.BI" LINE 1
2 CLEAR 32998
3 LET sort=33000
5 READ col: POKE sort-1,col
10 LET a=10: LET b=11: LET c=12: LET d=13: LET e=14: LET f=15
11 LET x=sort
12 READ a$: IF a$="END" THEN GO TO 15
```

BYTE POWER

Super Special!!!

All BYTE POWER titles at 50% off...

	.Now.
12 BYTE POWER Magazines (Back issues)	\$27.50
6 BYTE POWER Magazines (Back issues)	\$16.00
1 BYTE POWER Magazine (Back issue)	\$3.00
1st Class Pack (6 back issues in vinyl album)	\$20.00
Mega Pack (12 back issues in 2 vinyl albums).....	\$31.50

Issues Available...

August '86, September '86, October '86, November '86, December-January '87,
February '87, Spring '87, Fall '87, February '88, May '88, Fall '88, Winter '89

BYTE POWER 1st Class Fonts II	\$10.00
2+FAST (Fast Load for TS2068).....	\$10.00
The Print Factory	\$12.50
The Companion Volume 1	\$12.50
Best of Business and Utilities.....	\$10.00
Best of Arcade Games	\$10.00
Best of Entertainment, Strategy & Board Games	\$10.00

** For those with remaining issues of BYTE POWER Magazine, you may use them as credit towards the purchase of any of the above packages...

Please add \$3.00 for Shipping & Handling. U.S. Funds only...

**BYTE POWER
1748 Meadowview Avenue
Pickering, Ontario, Canada
L1V 3G8**

MINERVA and OTHERS

Some time back the QL group of the Toronto Timex-Sinclair Users Club met at the residence of the QL librarian " H. Howie " and we had a most enjoyable afternoon with the presence of Paul Holmgren. A number of people in attendance wrote about this session in the past few issues of Sinc Link . Some members took advantage of the occasion to order some hardware for the QL, e.g. the clock board . Others like myself ordered the MINERVA both for the cooler running QL and also the fact that a few bugs in the original version of the ROM had been corrected.

After a few delays I received my MINERVA with SPEEDSCREEN .The delays were caused by the fact that the list of requests had been misplaced.

Once the MINERVA was installed lo and behold a few of the games that I had obtained before and could not run were now running beautifully. For example DOMINOES is now functioning to perfection.

I must admit that in the process of installing MINERVA I also damaged the keyboard membrane and had to replace it. But that is another story.

An order was placed for a few membranes and at the same time I decided to take advantage of " SOLUTION ". I was more than pleased with the result of my decision. I can now branch out into MS DOS without the expense of buying a P C . I know that the QL with an emulator can never be as fast as an IBM PC or even a compatible, but with the price differential " I don't care !!! ". But I'll never again complain about the speed of the good old QL.

You too can take advantage of " SOLUTION " by contacting MECHANICAL AFFINITY either through Frank Davis at 513 East Main St. , Peru , Indiana , 46970 ; TEL : 317 473 8031 or Paul Holmgren at 5231 Wilton Wood Ct., Indianapolis , Indiana , 46254 ; TEL : 317 291 6002 .

At least get the catalog , the prices will amaze you !.

Louis Laferriere

Q L

2 0 6 8

In the last issue of Sinc-Link , Hugh Howie made a plea for programmers to give special considerations to the Canadian users of their programs . The problem is of course caused by the fact that our POSTAL CODES are composed of a combination of LETTERS and DIGITS. One programmer who has solved this problem is Jack Dohany with his *** MSM *** version of MSCRIPT. (2068) . There are three methods of sorting the file , using Z for ZIP CODE , L for LAST NAME and C for CATEGORY . The instruction MANUAL suggest using !!! e.g. EXCLAMATION POINTS to identify the WORD to sort on for the first and last line of the record. Very useful for CANADIAN users.

Louis Laferriere

Fixing My FDD3000

by
James F. Brezina

A while back, something I feared would happen, did. My FDD 3000 stopped working. Since Zebra Systems quit supporting the TS computers and dropped the FDD3000, I didn't know where I could get it fixed. Dan Elliot has never mentioned the repair of disk systems, I didn't know if I could send it to him. However, since my work before retirement was the repair of electronic equipment, I thought I would try to fix it myself. The big drawback was that I had no schematics on the unit. I had purchased the technical manual on the system, but, that was no help. All it contained was machine language and no schematics. Therefore, it was of no use to me in repairing the unit. I had to figure out how to disassemble the unit on my own. I was hoping that I did not have to remove the disk drives.

All owners of the FDD know that, immediately after turning it on and then inserting the disk, you hear the disk spinning and the indicator light flashes. Turning on the computer next causes the disk to spin again and the copyrights to appear on the monitor screen. My unit did neither. I presumed, from this, that the motor which spins the disk was not working for some reason.

I removed the cover of the unit and looked down at the power supply section. I could see 8 rectifier diodes and most of the rest of it was covered by a heat sink fastened to three voltage regulators. The power supply board was held down at the front by a moveable piece of gray plastic. The back end was under the transformer. I took my volt-ohm-milliammeter and checked the voltage on the diodes. They were all under 10 and 5 volts. Something was causing that voltage to be low and, I would have to remove the power supply assembly to work on it further.

The power supply assembly was mounted on a gray plastic frame which was mounted to the bottom of the cabinet with three screws. There is a wire from the transformer which is connected to the power fuse holder with a push on connector. You will have to pull this off the lug on the fuse holder or you will break the lug off the fuse holder. I didn't do this and had to replace the fuse holder. After removing the screws from the bottom of the cabinet, I found that I had to carefully move a slot in the plastic frame over the power line grounding lug in order to free the power supply from the cabinet. There were three 3 wire cables coming off the power supply board. They ran under the drives and had enough slack to let me remove the assembly from the cabinet enough to work on it. Where the wires went into the board, they were marked +5 and +12. I don't recall any marking for the third wire which had to be ground.

The heat sink on the regulators had threaded holes for screwing the tabs on the regulators to it. I was glad the screws weren't mounted with nuts. The transformer was held in place by formed corners in the plastic frame and by a plastic cover that was held in place by screws through the plastic frame. When I removed the cover and picked the unit up by the transformer, the power supply board fell off the lugs on the transformer. The solder on the power supply board showed no impressions that would indicate a secure solder connection to the lugs on the transformer. This was the cause of my trouble. It was also, in my opinion a manufacturers defect, but, since the vendor has deserted us, and it was out of warranty, I had no recourse to the vendor for warranty repair.

I removed the old solder from those holes and resoldered the transformer to the power supply board. I then replaced the heat sink on the regulators and reconnected the wire to the fuse holder. I set the power supply assembly on its side and reconnected the FDD to my computer. I turned the FDD on and checked the voltages coming off the diodes. They were now the required 5 and 12 volts. I placed a disk in drive A and heard the spinning sound. I turned the computer on, heard the spinning sound, and got all the copyright notices on the screen. I then tried loading in a program from the disk and that worked. I had repaired my FDD!



"I can't decide between the desktop or the laptop."



"Whoa! That was a good one! Try it, Hobbs—just poke his brain right where my finger is."

H E L P * * * Q L

+++++

For some time now I have been holding some programs given to me by Real Gagnon, formerly the EDITOR of QL_DOC from Montreal. The reason I didn't give the programs to H.Howie our esteemed QL Librarian was because the programs and files were written in GERMAN. The programs themselves run beautifully on my QL now that MINERVA is helping, except that the instructions are in GERMAN and the query for INPUTS are not easily understood by me. I have been trying to translate both the queries and instructions but even with the help of a large COLLINS dictionary it is a very slow and painful project. I had asked at one time for assistance from one of our members but he has since moved away, I believe to MONTREAL. The graphics of the programs are superb and so it is a shame not to take advantage of these programs.

If anybody can help in the translation, please get in touch either with the EDITOR, Jeff Taylor, Hugh Howie, George Chambers or myself and we will supply the necessary listing or disks in any format required.

Louis Laferriere
care of : SINK-LINK EDITOR

TORONTO TIMEX-SINCLAIR USERS CLUB

14 RICHMOND COURT,
SCARBOROUGH, ONTARIO,
CANADA,

M1K 2Y1

June 13, 1991

Mr. George Chambers
14 Richome Ct.
Scarborough, Ont.
Canada M1K 2Y1

Dear George,

Just thought I would drop you a line concerning one of your advertisers.

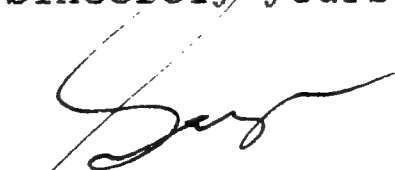
Unlike some software suppliers who take 9 months to deliver (I won't mention a name) and who don't acknowledge your order - and then like Jack Doheny who takes forever and a day but at least lets you know when to expect your order - - I have GOOD news!!

John McMichael, who had an ad in SINC-LINK May/June '91, should have a medal for promptness. Seven day turnaround from my mailing date until I received his Collections #1 & #2 by mail!! That doesn't end the story. Collection #2 had a bug in it and only two of the programs LOADED so I mailed it back to him along with another check for his Collection #3. (As I told him - I had faith!!) SIX days later I had a new copy to replace the Collection #2 and the new disk, Collection #3, with an apology about the bad disk. The replacement disk and the new disk LOADED perfectly and to say I was "happy" with the service would be an understatement!! What a joy that there are a few good suppliers out there that believe in good service.

For an investment of \$26.95 for the three disks, you get about 445 graphics of various sizes which LOAD into any of the PRINT FACTORY programs. True, they are not all new and original; but it is a great collection. Some are digitized and some are enlarged from small graphics and some are small graphics - each marked with a code so you can tell what will be coming up when LOADED. Very impressive. If you buy #1 & #2 at \$17.95, you can buy #3 for \$7 - all ppd.

Just thought maybe you'd like to hear some good news and maybe John will continue to put out programs like this if he is encouraged a little bit with our support.

Sincerely yours,



George G. Cary
P.O. Box 336
Coloma, CA 95613 USA

FOR SALE

2 TS2068 Computers
1 TS2040 Printer
1 TS2050 Modem (uncased)

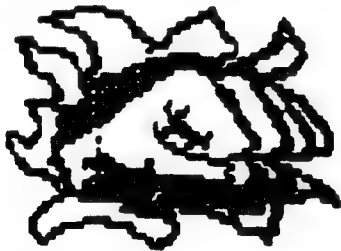
Bruce Lessard, 29 Hillside Ave.,
Salem, MA 01970

Bruce mentioned these items to me some
months ago. Call him at 508-741-0482
if interested, or write him. GFC

Put some **punch** in your
PRINT FACTORY® graphics!!

Collection#1 **Collection#2**

Miscellaneous
Men/Boys
Women/Girls
Digitized Misc.



Animals
Funny
Sports
Transportation
Computers
Food
Holidays
Xmas

*Over 150 large & small graphics
in each collection!*

Only \$12.95 ppd. each or both collections
for \$19.95 ppd. [Check or money order.]
Available on tape or 5 1/4 DSDD LARKEN
or OLIGER disk - please specify.

Send order to:
John McMichael, 1710 Palmer Dr.,
Laramie, WY 82070

UPDATE MAGAZINE
513 E. Main Street
Peru, IN 46970

NEWS AND

NEW

PRODUCTS

Quick Directory Sorter
by Steven V. Gunhouse

If you have seen any recent version of Omnibus, then you have seen my program "dsort", a Larken directory sort written in BASIC, and possibly the version Bob Mitchell compiled.

For some reason I haven't figured out, the compiled version doesn't work on my system - I have LKDOS for the AERCO disk interface. That, and it still seemed a bit slow, as far as I was concerned.

What could be done about it? There is one way to get quicker than a compiled program, and that is in machine code. Some things that are difficult in BASIC become almost impossible in ml, though, so I had to leave out the "Sort by NAME or TYPE?" option. This routine always sorts by name in alphabetical order.

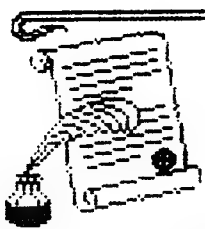
So type in the accompanying BASIC loader, and save it. Then anytime you want to sort your directory on a disk, run this routine. Insert the disk to be sorted, press a key, and its all done in under 10 seconds! That won't leave you a chance to go get a cup of coffee, but it may sort your whole library in under 5 minutes.

```

1 PRINT "Loading data...": LE
T a=10: LET b=11: LET c=12: LET
d=13: LET e=14: LET f=15
2 LET st=65200: LET x=st
3 READ a$: IF a$<>"END" THEN
FOR i=1 TO LEN a$ STEP 3: LET n
=16*VAL a$(i)+VAL a$(i+1): POKE
x,n: LET x=x+1: NEXT i: GO TO 3
5 CLS : PRINT AT 7,3;"Quick D
irectory Sort"
6 PRINT AT 10,0;"Sorts by NAM
E only!"
7 PRINT '"Insert disk to sort
. Be certain write protect is re
moved."'
8 PRINT '"Drive to sort (0-4,
Q to quit, any other key for c
urrent drive)"
10 LET a$=INKEY$: IF a$="" THE
N GO TO 10
11 IF a$="Q" OR a$="q" THEN P
RINT #4: GO TO 0: PRINT #4: NEW

12 IF a$>="0" AND a$<="4" THEN
PRINT #4: GO TO VAL a$
13 RANDOMIZE USR st
14 CLS : PRINT #4: CAT "",
15 PRINT '"Done - press any ke
y."
16 PAUSE 0: GO TO 5
1000 DATA "F3 CD 62 00 CD 96 00
AF"
1001 DATA "32 1D 20 CD 7E 00 CD
7B"
1002 DATA "00 21 88 20 3E FF 47
23"
1003 DATA "BE 20 FC 78 23 BE 20
FC"
1004 DATA "11 BF 34 E5 13 23 7E
12"
1005 DATA "FE FA 28 03 B8 20 F5
4F"
1006 DATA "78 12 22 06 20 E1 3A
C1"
1007 DATA "34 A7 28 36 E5 11 BF
34"
1008 DATA "2B 7E FE F5 28 19 B8
20"
1009 DATA "F7 E5 13 23 1A BE 28
FA"
1010 DATA "E1 30 0C D1 E5 23 13
7E"
1011 DATA "12 B8 20 F9 E1 18 DD
D1"
1012 DATA "21 BF 34 23 13 7E 12
B8"
1013 DATA "20 F9 2A 06 20 79 77
B8"
1014 DATA "28 AE CD 78 00 C3 BA
00"
1015 DATA "END"
9999 PRINT #4: SAVE "QDSort.B1"
LINE 1

```

BOB'S Notebook

by:
Bob Mitchell
20 Uild Briarway
WILLOUDALE Ontario
M2J 2L2

May 91

TOOLKIT: PART2 INFORMATION & INTERRUPT OPTIONS

This is the second in a series of tutorials designed to give TS2068 users some guidance on using the various TOOLKIT options. This article will focus on those functions using the Diverted Interrupts feature. These are Disable New, Autoline Number, Constant Memory, Trace and finally Return Interrupts to Normal. It will also cover the INFORMATION functions.

Once again, it is very wise to SAVE your work many times as you proceed, for as Murphy is quoted, "If something CAN go wrong, it WILL". If you have the interrupts turned ON (ie, diverted), do not try to make a LARKEN AUTOSTART SAVE of any listing you are working on. Check toolkit option I; if the interrupts are diverted, press the X option on the menu to make them normal. Then go ahead with the AUTOSTART SAVE.

Before going into the "Interrupt" options, let's cover the INFORMATION or STATUS routines: INFORMATION, LIST VARIABLES, HEX & DEC conversions.

INFORMATION

This will reveal some useful information:
Amount of memory left: Note this is five bytes off the mark by virtue of the different TS2068 ROM being used.
Interrupts Status: Diverted or Normal. These could be referred to as ON or OFF respectively.
User Defined Graphics display. These can be completely garbled but can be restored to their normal status by using the UDG Designer. More on this in a later article.
Plus some others: Program length & Variables length which added to Memory left = User RAM available. A command can be entered without returning to the menu.

LIST VARIABLES

Displays, in the order declared, all simple variables with their values except that CHR\$ 0 to 31 in strings are printed as <?> (thus z\$="??text??" would imply some attributes included in the string to give colour to text). FOR-NEXT variables are shown in CAPITALS. DIM variables are printed with their dimensions but no values.

The variables shown will be those in the VARS area. Just because variables are shown in the listing does not mean that they are alive. It is best to check this option when all or most of the variables in any program have been exercised, or at the very least be aware that all variables in the listing may not be displayed.

HEX & DEC

Very useful display of both hex and decimal values for any number (0-65535); you can keep on entering values and exit to the menu by pressing ENTER.

By way of explanation, it is helpful to understand the ROM "collect number routine" which uses the ROM editor to allow input up to five bytes; these must be positive whole numbers. If the first character is <#>, the characters A-F or a-f are treated as digits and a hex number is evaluated.

Now the INTERRUPT OPTIONS:

DISABLE NEW

This is the key routine for Diverted Interrupts. Functions <N>, <Q> and <T> are all routed through here so that when THEY are ON, the <W> function is ON too.

Pressing <A> in the K Mode will give STOP and not NEW, a safety feature when programming. If you press Symbol Shift and A in the L Mode, nothing appears to happen; but the next key pressed will be taken as K Mode. Symbol Shift A followed by just A will give STOP. This is useful when using Search and List or Search and Replace as it avoids having to THEN <keyword> <backspace> DELETE <forward space>, bad enough in Spectrum but a gigantic pain in the neck with the TS2068. More on this when we come to those options in a later article.

☐ AUTOLINE ON

This is best used when you are developing a new program. You decide on the increment (usually 10 but not more than 100). Once this option is activated and each time the ENTER key is pressed, a line number will be inserted in the EDIT line. This line number will be the sum of the current line and the increment. You may change the line number at any time by DELETING it and inputting a new one less the increment (eg, if you want to start numbering lines at 9000 and the increment is 10, input 8990).

It will very rarely append to a line after a syntax error and you will have to delete such an error.

This routine will be turned OFF if Constant Memory, Trace, Disable New or Interrupts Normal routines are turned ON. Use RANDOMIZE USR 60000 to get to the menu and then press <x>.

☐ DISPLAY MEMORY

This will display Memory Left in the upper right hand corner of the screen. The value will alter as the free memory changes.

This routine will be turned OFF if Autoline Number, Trace, Disable New or Interrupts Normal routines are turned ON.

☐ TRACE ON

This gives a constant display of the current program line. First set the speed (max 100 is close to normal running speed). Reducing the number will slow down the response time so that you can get a rough idea why a program is going wrong and at what line. You can override the speed set by using <ENTER and K together> or freeze the program by holding down <ENTER and L together>.

This routine will be turned OFF if Autoline Number, Constant Memory, Disable New or Interrupts Normal routines are turned ON.

☒ N/Q/T/W OFF (Interrupts NORMAL)

Switches off any and all the interrupt options. You can now use NEW if you wish!

```
.....
This is a sample Status Report
using <I>. Note: interrupts
are diverted, even though no
interrupt options may have
been activated; also that
memory left is five bytes over
the PRINT FREE value.
.....
```

```
STATUS REPORT
MEMORY LEFT: 28042 bytes
PROGRAM LENGTH: 10401 bytes
VARIABLE FILE: 332 bytes
USER RAM: 38775 bytes
INTERRUPTS DIVERTED
```

```
.....
A sample Variables List
prepared for this article by
saving it to a RND/SEQ file
using the Larken command:
<RANDOMIZE USR 100: OPEN #2,
"NAME OUT">. Stream 2 had
previously been OPENed to
channel "p" ready for this.
.....
```

```
p0=1
oz=8200
oy=110
oo=0
oa=1
p$=Put           in Drive 0,
then press a key"
d=2
N=5
h$=""
d$(12)
n$="e"
g$="??PICA?? "
```

323 1/2 N. Church Street
Bowling Green, OH 43402
May 6, 1990

Dear George,

Well, it's May, so it must be time for me to write again. Not quite; I haven't gotten my May Sinc-Link yet. But I did receive my May issue of Byte recently, and it had something I thought was so interesting I just had to try it.

Have you ever noticed just how bad the CIRCLE command is on our computers? It is very slow, and especially for small circles it isn't even accurate. In this month's Byte they had an article on graphics for the IBM, and included a listing in the Pascal language of an efficient circle routine. I decided I just had to convert it to our machine code and try it out.

Actually, I made a few changes to try and optimize it for ml, so it isn't strictly the same as the original. I also threw in error checking. And to save a few POKEs, I made it use the current PLOT position as its center. Setting the radius still requires a POKE, but that didn't seem too bad.

As you might expect, I did have the computer test the accuracy of this routine. As near as practical, it is totally accurate. That is to say, it rounds off the same way as a BASIC program using SQR to compute values would. That means, it draws a circle of radius 1 as a small square, but those are the limits of the display.

And the routine is fast. Since it doesn't actually use any SQR or SIN and COS, it doesn't use the ml calculator routines at all. And it will draw a large circle, as large as the screen will hold, so fast I can't see it working. It just appears on the screen.

I can see that this might be useful for people who want to work in graphics with their programs. It is only good for circles, of course, but it is fast enough to do a reasonable imitation of a bouncing ball, for example.

I set up the routine to leave the PLOT position unchanged. So for instance, if you wanted to draw several circles with the same center, all you have to do is POKE the new radius and call the routine. To set colours, you will have to use the standard commands, and then change them back when you are done.

I am not about to try and explain how it works, if you want to know, pick up the May, 1990 BYTE magazine and read page 282.

That is it for now. I may have something else in a couple of weeks, after I get this month's Sinc-Link. But until then, take care.

Oh, I had time to work on it because college is over already. I should still be here for the foreseeable future, though. Bye for now.

Sincerely,

Steven V. Gamhouse


```

1 REM Hex Loader for Circle
2 REM with demo
3 CLS : PRINT AT 10,5;"Reading Circle CODE"
4
5 LET A=10: LET B=11: LET C=1
6 LET D=13: LET E=14: LET F=15
7 LET x=23300
8 READ a$: IF a$="End" THEN GOTO 10
9 FOR i=1 TO LEN a$ STEP 3: POKE x,16*VAL a$(i)+VAL a$(i+1): LET x=x+1: NEXT i: GOTO 8
10 CLS : PRINT "Demo using CIRCLE Command:"
11 FOR i=3 TO 87 STEP 3: CIRCLE 128,88,i: NEXT i
12 PAUSE 30
13 CLS : PRINT "Now using the Circle ml:"
14 FOR i=3 TO 87 STEP 3: PLOT INVERSE 1;128,88: POKE 23299,i: RANDOMIZE USR 23300: NEXT i
15 STOP
1000 DATA "2A 7D 5C 4D 3A 03 5B 57"
1001 DATA "7D BA 38 60 82 38 5D 7C"
1002 DATA "BA 38 59 82 38 56 FE B0"
1003 DATA "30 52 47 7A 1E 00 CB 22"
1004 DATA "F5 C5 E5 D5 C5 CD 3E 26"
1005 DATA "C1 D1 78 92 B8 28 08 47"
1006 DATA "D5 C5 CD 3E 26 C1 D1 79"
1007 DATA "93 B9 28 13 4F D5 C5 CD"
1008 DATA "3E 26 C1 D1 78 82 B8 28"
1009 DATA "06 47 D5 CD 3E 26 D1 E1"
1010 DATA "C1 F1 1C BB 28 02 30 04"
1011 DATA "05 15 82 15 BB 28 06 38"
1012 DATA "04 0C 93 1C 1C 1D F5 78"
1013 DATA "BC 30 B6 F1 22 7D 5C C9"
1014 DATA "End"

```

For the Bouncing Ball demo, copy the previous program except lines 10 to 16 (or type DELETE 10,16) and enter:

```

2 REM with bouncing ball demo
10 CLS : BORDER 6
11 LET r=10: LET x=r: LET y=r: LET d=2: LET e=17
12 OVER 1: POKE 23299,r
13 PLOT INVERSE 1;x,y: RANDOMIZE USR 23300
14 IF x+d<r THEN LET d=ABS d
15 IF x+d+r>255 THEN LET d=-d
16 IF y+e<r THEN LET e=ABS e: GOTO 18
17 LET e=e-1
18 LET x=x+d: LET y=y+e: RANDOMIZE USR 23300: PLOT INVERSE 1;x,y: RANDOMIZE USR 23300
19 GOTO 14

```

YOUR AD COULD

GO HERE.

ADVERTISE

FOR FREE IN

SINC-LINK

MICRODRIVE PROBLEM SOLVED?

(ONE ANYWAY---maybe)

NEVER OVER-WRITE

We all know how we feel when we get that ill_famed well-known message, "Bad or Changed Medium" Well there could be a cure to some extent for it.

For some time now when I have been using cartridges and have had to overwrite something when saving, I have made it a habit as soon as the OVER-WRITE? message appears, to DELETE that which is to be overwritten, and save as normal.

I have no idea how this works but it does. Since using this method I have reduced my BAD messages considerably. So why not give it a try. It works for me so it should work for you.

The amazing thing is that just after I wrote this, the May issue of QUANTA was delivered, and in it there is a letter from Mr R. Gilbert of Dartmouth, Nova Scotia, and he says that the problem is that sometimes part of a program can be written to a sector which is already in use!

He does not over-write any more either. He saves to a new cartridge. Similar to what I have found.

By the way, If you are a Qler and do not get QUANTA, you should.

H.H.H.

ZX80 SCHEMATIC

Does any club member have a schematic for the Sinclair ZX80 computer. I have had a request for just this item.

*George Chambers, 14 Richome Court
Scarborough, Ont M1K 2Y1*

LOOKING FOR A TS1500

I have had someone contact me, looking for a TS1500. I said that I did not now of any available. Do you have one to sell? GFC.

HELP!!!!!!

I HAVE BEEN SEARCHING FOR A BASIC LISTING FOR A RAILROAD SWITCHING PROGRAM FOR SOME TIME NOW WITHOUT ANY SUCESS.

THE TYPE OF PROGRAM I AM LOOKING FOR IS AVAILABLE COMMERCIALY FROM SEVERAL SOFTWARE DISTRIBUTORS BUT IS FOR COMPUTERS OTHER THAN OUR T/S 2068.

I AM NOT REFERRING TO A TRAFFIC CARD GENERATING PROGRAM BUT TO ONE WHICH WILL ALLOW THE USER TO SIMULATE GRAPHICALLY A TRACK LAYOUT AND ENABLE A LOCOMOTIVE AND CARS TO BE RUN OVER THIS TRACKAGE, SWITCHING CARS FROM LOCATION TO LOCATION, COUPLING AND UNCOUPLING THE CARS AS DESIRED, THE OBJECTIVE BEING TO SOLVE VARIOUS SWITCHING PROBLEMS IN THE LEAST NUMBER OF TRAIN MOVES.

IF I CAN FIND A PROGRAM LISTING IN BASIC I WANT TO ATTEMPT TO CONVERT IT TO TIMEX/SINCLAIR BASIC. THIS IS FOR MY OWN USE (OR OTHERS IF INTERESTED) AND IS IN NO WAY A COMMERCIAL VENTURE.

THANKS,

DAVE SANDULLI
205 OLD TOWN FARM RD.
WOODBURY, CT 06798
USA

WHAT PRICE SUCCESS?

DO WE NEED A CENTRAL ORGANISATION?

by Hugh H Howie

What happened to ????? Magazine? What happened to the ????? club? Why did ????? not get off the ground?

How often have we heard questions similar to this being asked. How often have we turned to ourselves for the answer?

Why did SNUG not get any further than it did in all this time? The answer is not known to all of us, perhaps it is not known to any of us. The last count I heard was that they had about 130 members. This is enough to allow a News Letter to get off the ground, so it was not lack of members that held it up. After all this time with no News Letter being produced, is it surprising that many did not renew their subscriptions? I don't think so. I for one would not renew where there was nothing for my money.

Now we have a new outfit coming along which promises to take up where SNUG left off. This new outfit is called TSNUG, and is going to produce a News Letter called "ZXir Qlive Alive" What a name, but at least with some imagination. The promises coming from this group are very nice, but can they produce? Time alone will tell, and also how many subscribers they get will also be a factor; probably the big factor.

Do we really need another organisation to look after our interests is the most prominent question in the mind of most of us. Can they do anything more for us than we can for ourselves? Certainly not! It is up to one and all of us to do what we can for our Sinclair Product. It is up to me to look after MY interests. But we must remember that we also need an organisation to meld us all together into one reasonably cohesive unit. Tsnug can be that organisation. But it must have support from us, the end user who will derive the benefits of such organisation, to assist in the dissemination of knowledge. Goodness knows we need all the help we can get in our hobby.

To sit back and say we will wait and see how it gets on before we spend our dollar is just the same as burying our heads in the sand and wiggling our tails at the stars. If no subscriptions come in till we

see how things go, then nothing will go. Subscriptions are needed to get started, and to continue, a lot of interest is going to have to be shown. Not only by subscribers but by contributors. If we want another Sinclair oriented News Letter and central organisation we must participate from the word GO!

Of course we could also take a look at our own Sinc-Link. This is one of the better ones on this continent. We, if we so desired could perhaps step in here and form a central group. One thing for sure is that we already have an excellent Letter to start with. One which is published On-Time Every-Time, With a diversity of views and news unsurpassed by any.

We have a core of excellent contributors in many subjects, and I feel that with very little effort this core could be increased. We could ourselves be the leaders. But it would require a lot more work on the part of many of us. We would all have to put our shoulder to the wheel and fingers to keyboard to ensure success.

Then again, if we were to even think of going this route, then why are so many of us sitting on our skyward pointing tails doing nothing for ourselves. What we need is a good dose of that stuff that makes you git up an' go! Call it what you may, but I think a good name for it would be PRIDE. We should have more pride in our Sinclair Computers. More pride in our Club. More pride in our future. The main thing is we need more pride in our own abilities. We must not stand still. We must go forward.

This will not come about by waiting to see how things go. It will only come about by action. Be it supporting another group from the beginning, or supporting the group to which we belong, no matter what that group be.

Nothing can prosper without support. Nothing can prosper if we all wait and see. If you WANT action, you must TAKE action.

GO FOR IT

Bob's Notebook

In response to remarks that the listing in Sinc-Link Jan-Feb 91 of Character Set Editor had some errors and was generally hard to enter, here is a second listing. This listing was done on the TS2040 printer using a heavy font to make it easier to type in. Use the earlier listing for guidance in entering the graphic and INVERSE characters.

```

100 REM CHARACTER SET EDITOR
110 REM By Bob Mitchell 1990

120 RESTORE 8500
130 FOR i=0 TO 41: READ n0: POKE
E 63163+i,n0: NEXT i
140 POKE 23658,0: POKE 23607,60
: BORDER 0: PAPER 0: INK 7: OVER
0: CLS
150 LET j$=" !""#$%&'()*+,-./01
23456789:;<=>?@ABCDEFGHIJKLMNPQ
RSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{ }~"
160 LET b=1
170 LET po=60
180 DIM c(12): DIM h(8)
190 LET h(1)=128: FOR f=2 TO 8:
LET h(f)=h(f-1)/2: NEXT f
200 BORDER 0: PAPER 0: INK 7: C
LS: PR"IT "Character Set Editor
""": NEXT: PRINT ""1 > DE
FINE set""2 > SAVE/LOAD set""
"3 > ROTATE whole set"
210 LET i$=INKEY$: IF i$<"1" OR
i$>"3" THEN GO TO 210
220 IF i$="1" THEN GO TO 1000
230 IF i$="2" THEN GO TO 2000
240 IF i$="3" THEN GO TO 4000
1000 PAPER 1: OVER 0: BORDER 1:
INK 9: CLS: PRINT AT 0,0;"1 > E
ditor"
1010 FOR f=1 TO 8: POKE USR "a"+
f,129: NEXT f: POKE USR "a",255:
POKE USR "a"+7,255
1020 POKE 23607,60: PRINT AT 15,
0: INVERSE 1:j$(1 TO 32)""j$(33
TO 64)""j$(65 TO 96)
1030 LET ch=251
1040 POKE 23607,ch: PRINT AT 16,
0:j$(1 TO 32)""j$(33 TO 64)""j$(
65 TO 96)
1050 OVER 0: INK 7
1060 POKE 23607,po: INPUT AT 0,0
;"Character to be edited? "; LI
NE c$: IF CODE c$>127 OR CODE c$
<32 OR LEN c$>1 THEN BEEP .5,-20
: GO TO 1060
1070 PRINT AT 13,0: INVERSE 1;"E
DITING": c$
1080 FOR f=1 TO 8: PRINT AT f,0:
"AAAAAAAAA AAAAAAAAA": NEXT f
1090 PRINT AT 0,0: INK 2: PAPER
7;"COMPARE 87654321"
1100 LET x=1: LET y=9: DIM a(8)
1110 DIM b$(8,8): LET k=(CODE c$
-32)*8

```

```

1120 PRINT AT 0,18;"Cursor keys
to";AT 1,19;"move.";AT 2,17;"p=p
lot.";AT 3,17;"e=erase.";AT 4,17
;"c=store chr."
1130 PRINT AT 5,17;"n=store chr.
";AT 6,18;"and return";AT 7,18;"
to menu.";AT 8,17;"o=compare";AT
9,18;"chr.";AT 10,17;"q=overlay
a";AT 11,18;"chr.";AT 12,17;"m=
facility menu"
1140 PLOT 135,104: DRAW 0,-38: D
RAW 120,0
1150 OVER 1: INK 5: PRINT AT x,y
;"": BEEP .003,x+y: PRINT AT x,
y;" "
1160 LET i$=INKEY$
1170 IF i$="8" AND y<16 THEN LET
y=y+1
1180 IF i$="5" AND y>9 THEN LET
y=y-1
1190 IF i$="7" AND x>1 THEN LET
x=x-1
1200 IF i$="6" AND x<8 THEN LET
x=x+1
1210 IF i$="p" AND b$(x,y-8)=""
THEN PRINT AT x,y: OVER 0;"":
LET b$(x,y-8)=""": LET a(x)=a(x)
+2+(16-y)
1220 IF i$="e" AND b$(x,y-8)=""
THEN PRINT AT x,y: OVER 0;"A":
LET b$(x,y-8)=""": LET a(x)=a(x)
-2+(16-y)
1230 IF i$="o" THEN GO SUB 1300
1240 IF i$="c" THEN GO SUB 1290:
GO SUB 1440: GO TO 1050
1250 IF i$="n" THEN GO SUB 1290:
GO SUB 1440: GO TO 1030
1260 IF i$="q" THEN GO SUB 1370
1270 IF i$="m" THEN CLS: BORDER
0: PAPER 0: OVER 0: INK 7: GO T
O 200
1280 GO TO 1150
1290 OVER 0: FOR f=1 TO 8: POKE
ch+256+256+k+f-1,a(f): NEXT f: R
ETURN
1300 INPUT "Chr to be compared?
"; LINE g$: IF CODE g$<32 OR CO
DE g$>127 OR LEN g$>1 THEN BEEP
.5,-20: GO TO 1300
1310 FOR f=1 TO 8: PRINT AT f,0:
OVER 0;"AAAAAAAAA": NEXT f
1320 FOR f=0 TO 7: LET gr=PEEK (
ch+256+256+((CODE g$-32)*8)+f)
1330 FOR g=1 TO 8
1340 IF gr>=h(g) THEN PRINT AT 1
+f,g-1: OVER 0;"": LET gr=gr-h(
g)
1350 NEXT g: NEXT f
1360 RETURN
1370 INPUT "Chr to be overlaid?
"; LINE g$: IF CODE g$<32 OR CO
DE g$>127 OR LEN g$>1 THEN BEEP
.5,-20: GO TO 1370
1380 FOR f=1 TO 8: PRINT AT f,0:
OVER 0;"AAAAAAAAA": NEXT f
1390 DIM a(8)
1400 FOR f=0 TO 7: LET gr=PEEK (
ch+256+256+((CODE g$-32)*8)+f)
1410 FOR g=1 TO 8
1420 IF gr>=h(g) THEN PRINT AT 1
+f,g+8: OVER 0;"": LET gr=gr-h(
g): LET a(f+1)=a(f+1)+h(g): LET
b$(f+1,g)=""
1430 NEXT g: NEXT f: RETURN
1440 LET k=CODE c$-32: IF k<32 T
HEN LET xx=16: GO TO 1470
1450 IF k<64 THEN LET xx=18: LET
k=k-32: GO TO 1470
1460 IF k<96 THEN LET xx=20: LET
k=k-64
1470 POKE 23607,ch: PRINT AT xx,
k:c$: POKE 23607,po: RETURN

```



```

2000 INPUT "1=SAVE 2=LOAD ";s1
2010 IF s1 THEN INPUT ("SIDE fil
e name (<=5 "); LINE n$
2020 IF NOT s1 THEN INPUT ("CODE
file name (<=5 "); LINE n$
2030 INPUT "drive? ";drv: RANDOM
IZE USR 100: GO TO drv
2040 IF s1 THEN RANDOMIZE USR 10
0: SAVE n$+"."CF"CODE 64512,768
2050 IF NOT s1 THEN RANDOMIZE US
R 100: LOAD n$+"."CF"CODE 64512
2060 BEEP 1,10: GO TO 200
4000 REM rotate font
4010 INPUT ("ROTATE CLOCKWISE"
1=1/4 Turn 2=1/2 Turn 3=3/4
Turn "); LINE r$
4020 IF r$("1" OR r$)"3" THEN BE
EP 1,1: GO TO 4010
4030 LET rot=CODE r$-48
4040 INPUT "; PRINT #0;"Stand by
""
4050 FOR r=1 TO rot
4060 FOR q=0 TO 95
4070 LET l=64512+q+8
4080 RANDOMIZE l: POKE 63165,PEE
K 23678: POKE 63167,PEEK 23671
4090 RANDOMIZE USR 63163
4100 NEXT q
4110 NEXT r
4120 BEEP 1,10: INPUT "; PRINT #
0;"Rotation finished." "Press a
key.": PAUSE 0: GO TO 200
4130 STOP
8500 DATA 38,128,33,0,226,6,1,14
8510 DATA 0,229,123,166,254,0,40
,3
8520 DATA 121,128,79,203,32,35,4
0,242
8530 DATA 225,197,203,59,48,231,
17,7
8540 DATA 0,6,8,25,209,115,43,16
8550 DATA 251,201
9000 INPUT "SIDE PROGRAM drive?
";drv: RANDOMIZE USR 100: GO TO
drv: RANDOMIZE USR 100: SAVE "re
def.B1" LINE 100

```

CORRECTION!

Page 4 of the May-Jun issue of
Sinc-Link

Under Explanations against line
2580: LK008 address 16100
should read LK008 8214.

VU-CALC TIP

Line 3200 should start with
CLEAR VAL '29327' to ensure the
program will calculate after a
CLEAR WORKSHEET is given.

When a data file is loaded, you
may get a crash when you try to
calculate unless this change is
made.

Well, that is my experience, any
ay

Bob Mitchell 910524

ZX81 - PRINT USING

The "PRINT USING" article in this
newsletter has been excerpted from a
125-page paperback book called "The
Art Of Programming the 16K ZX81. It
is typical of the contents of this
book. The book is one of a series on
Sinclair computers that were
published by Bernard Babini
(publishing) Ltd.

Others in this series were "The
Art of Programming the 1K ZX81", "The
Art of Programming the Spectrum", "An
Introduction to Programming the QL",
"An Introduction to Z80 Machine
Code". They also publish a host of
other 'project' books, such as "Radio
Control for Beginners", "Model
Railway Experiments", "How to
identify Unmarked IC's", and maybe a
hundred others.

The books covering the Sinclair
computers are, of course, quite old,
having been published in 1984 and
thereabouts, and are likely to be out
of print. However if you wish to see
what is still available you could
inquire of your local bookseller.
Babini may have a Can/USA agent. Or
write directly to their address :
Bernard Babini (publishing) Ltd
The Grampians
Shepherds Bush Road
London W6 7NF
England

Ask about the BP-series of books.
They mention they will send a
catalogue of their range of
electronic books if you enclose a
stamped, self-addressed envelope. I
suggest that in lieu of stamps that
you send an International Reply
Coupon, available at any post office,
Canada or USA.

George Chambers

ZX81 - NUMBER FORMATTING PRINT USING

Most print formatting problems can be solved using a combination of truncating, rounding, or aligning the decimal point. However other versions of Basic have a very powerful statement PRINT USING that allows a wide range of number formats to be specified. It is possible to write a sub-routine that provides some of the capabilities of the PRINT USING command and this would be useful both for converting programs and for writing new programs.

The format of a number produced by a PRINT USING statement is specified by the use of a "picture" of the number stored in a string. For example, in most Basics "###.##" would specify a format of three spaces or digits in front of the decimal point and two digits following, i.e. 3.123 printed using that format would be blankblank3.12. There are many other formatting symbols that can be combined with # to form a "picture" of the number, but perhaps the most useful is the "floating" money sign. If you write either a dollar or a pound sign in front of the formatting "picture", the money sign will be printed to the immediate left of the formatted number.. For example, "###.###" would format 3.1234 as blankblank3.123. This method of drawing a "picture" of the number is a very easy to use and powerful formatting method. For example, if you don't want a decimal point printed then all you have to do is leave it out of the "picture", i.e. "###". If the number to be printed is too big for the space allocated to it by the "picture" then it is printed unformatted.

A general PRINT USING routine for the ZX81 would be rather long, but we can produce a subroutine that will accept a "picture" involving digit positions marked by an #, the decimal point and floating money signs. The only change that we have to make to the usual PRINT USING is to change the # to * because the ZX81 doesn't have a # character. The PRINT USING subroutine and a small test program is:

```
10 LET U$="*****.***"
20 PRINT TAB 9;U$
30 INPUT V
40 GOSUB 2000
50 LET V=V*100*RND
60 PRINT
70 GOTO 20

2000 LET H$=STR$ INT V
2010 LET L$=(STR$(V))(LEN H$+1 TO)
2020 IF L$<> "" THEN IF L$(1)="." THEN LET
L$=L$(2 TO)
2030 LET S$=U$(1)
2040 IF U$(1) <> " " AND U$(1) <> "$" THEN
LET S$=""
2050 LET F=0
2060 LET M=0
2070 LET N=0
2080 FOR I=1 TO LEN U$
2090 IF U$(I)="." THEN LET F=1
2100 IF U$(I)="*" AND F=0 THEN LET M=M+1
2110 IF U$(I)="*" AND F=1 THEN LET N=N+1
2120 NEXT I
2130 IF M=0 AND H$="0" THEN LET H$=""
2140 LET H$=S$+H$
2150 IF LEN H$>M THEN GOTO 2170
```

```
2160 LET H$=" "(1 TO M-LEN H$+H$
2170 IF F <> 0 THEN LET H$=H$+"."
2180 LET L$=L$+"0000000000000000"
2190 IF N <> 0 THEN LET H$=H$+L$(1 TO N)
2200 PRINT H$;
2210 RETURN
```

The "picture" is stored in the string U\$ at line 10. Lines 20-70 simply send test values in V for subroutine 2000 to format. Before any formatting begins the number contained in V is converted into a string by STR\$ and then split into two parts. The digits in front of the decimal point are stored in H\$ by line 2000 and the digits following the decimal point are stored in L\$ by lines 2010-2020. Notice the use of IF...THEN, IF...THEN construction in line 2020. This has the same effect as IF....AND...THEN, but is needed because in this case the second condition, i.e. L\$(1)=".", can only be worked out if the first condition is true i.e. if L\$ <> "". The expression IF L\$<> " " AND L\$(1) THEN... will give an error message if L\$ is null because in this case L\$(1) doesn't exist. The variable S\$ is used to hold any floating money sign in the formatting string U\$; if there is no such sign then S\$ is set to the null string (Lines 2030-2040). Lines 2050-2120 count the number of digits before the decimal point (M) and the number of digits after the decimal point (N) in the formatting string U\$. The variable S is zero if no decimal point is found. Line 2020 removes the leading zero if the number is less than one and there is no digit position specified by the format. Line 2030 adds the money sign, if any, to the front of the number. The "padding" blanks are then added to the number by lines 2150 and 2160. If a decimal point is required it is added in line 2170 and then the fractional part stored in L\$ is padded with trailing zeros (line 2180) before being truncated to fit into the correct number of digits by line 2190. Finally, line 2200 prints the fully formatted number.

Three samples illustrating different formats:

```
.****
.0001
.0086
.7543
30.9278
2318.9732
54588.0130
3579173.1000
62866000.0000
```

```
*.****
0.0001
0.0006
0.0056
0.2472
23.0431
2067.3304
59288.7480
3023603.1000
```

```
*****.**
0.01
0.43
18.18
993.24
96608.67
9170571.30
```

QL PRIMER

Ken Goods

Recently someone asked me what was the most important add-on accessory for the QL. This really got me thinking, in other words he was asking me to pick and choose between all the things I have come to take for granted. Could I actually do without my Trump Card and dual discs? Could I go back to 64 columns on a T.V. screen? How about giving up QRAM or Super ToolkitII? What would I then be stuck with? Answer; a great little machine that's reasonably priced, (make that downright cheap!), has it's own storage devices, and comes with four packaged programs that I am still not taking full advantage of.

The question was on my mind long after the person who asked it left without an answer. I wouldn't want to do without any of the luxuries that I have come to expect to be there when I boot up. But if I HAD to choose the one that I consider most valuable it would have to be Super Toolkit II. This is assuming that I already had a monitor. I can't imagine anyone who has 80 columns built-in not to take advantage of it, especially when a green mono monitor can be had for less than \$50.00. So, considering the monitor a gimme Super Toolkit II has to be the one.

Sinclair's SuperBasic is one of the best basics I know of. In addition to the "normal" basic commands there are the extended commands that all basic's do not have. I'm thinking back to the days when I was programming my trusty TS 1000 and agonizing over not having a RENUM command. How nice it would have been just to renum 100,10 and have a nice neat program with line numbers starting at 100 and going up by 10's. It makes it look like you never had to add a line, even if it wasn't that important anyway, (like 105 NEXT I). Not that I'd ever forget a NEXT I. But the real nice thing about SuperBasic is the Define procedure and Define Function commands. These commands work more or

less like a Gosub only once defined they are part of the operating system! This means you could define a procedure named print_variables in your program and if some reason the program has an error and kicks you out to basic, you could enter "print_variables" and all your variables would be printed to the screen and you could then figure out what that damned computer did to screw up your perfect program that you have been up half the night writing. Not only that, but, you can also re-define the commands that are already part of SuperBasic. Here's a little trick, want to keep people from seeing your program? Just re-define LIST, COPY, and EDIT, to NEW:

```
30000 DEFine PROCedure LIST
30010 NEW
30020 END DEFine Procedure LIST
```

Now if someone was to try to list your program it would execute the NEW command. Just do the same for COPY and EDIT and no-one will be able to see that sloppy code! (Unless they're very tricky and copy direct to the screen from a micro or floppy without first LOADING your program.)

As you can see, I like SuperBasic. But even so, it does have a few shortcomings. For one thing it doesn't have a PRINT USING command. First I didn't have RENUM in TS 1000 Basic and now this! Well here comes Super Toolkit II to the rescue. Just enter TK2_EXT and all at once you have 108 added commands to SuperBasic to make life much more simple. You also have a much more manageable editor, (full screen no less). There are so many handy commands included I can't possibly list them all, but I can give you an idea of how useful this software can be by explaining a few.

QL cont...

ALTKEY: This command allows you to "key define" commands. I normally define the ALT L combination as "load", ALT S as "save" etc. This saves a lot of typing, it's like having the trusty old TS 1000 single key commands again! I have several altkey define statements in my boot programs and they stay resident until I reset my QL. I hate having to print "PRINT #3," when I want something to go to the printer or, "PRINT #4," when I want something to go to a disc file. Now I have the number "1" on my keyboard altkeyed to "PRINT #1,", and so on. So, now when I'm programming I can just hit ALT 3 and my screen prints "PRINT #3,". Yes, it even works when you're writing a program in basic. Not only that, but it also works in Quill. If you find yourself writing the same word several times in a document just altkey it and it will be there with one combination keypress!

There are several commands that work with the default directories. Default directories can be set for EXecutable program, data file, and destination directories. If you wanted to copy "MYFILE" from flp1_ to flp2_ and your destination directory is flp2_ with your data directory being flp1_ all you have to do is print "COPY MYFILE" and it's done! This beats "COPY FLP1_MYFILE TO FLP2_MYFILE" huh? Of course, a little thought is in order here to make sure your programs are all set up the same way. Once you get used to it there's no other way to do it. After all, aren't we using computers to save time and effort? Another example; set your destination to be ser1 with your data directory still being flp1_ and if you want to list several programs all you have to do is "COPY XXX" where XXX is the file name. This sure beats "COPY FLP1_XXX TO SER1". A real time saver if you're listing more than a couple programs.

Something else that I've found very handy is the prompt you get when you try to copy a file to a device that already has a file by the same

name. In Superbasic you would get the error "already exists". If you were trying to backup this file, you would have to delete the old one and then copy it. With Super Toolkit II you get a prompt that says, MDV1_XXX already exists, overwrite? Y/N/A/Q. These mean YES, NO, ALL, QUIT. All you would have to do is press Y and your QL will happily go about copying the file right over the top of the old file. If you're wondering what the ALL response is for, there are also wildcard commands at your disposal, one of them being WCOPY. Now, if you were backing up a whole disc you could enter WCOPY FLP1_ to Flp2_ and the same prompt would come up (since your backup disc already has old versions of the same programs you're trying to backup). Now you probably know what the ALL command is for! You catch on fast! This time you answer with A (for ALL) and walk away, the WCOPY will take over and copy all the files from flp1_ to flp2_ without asking if it's alright to overwrite them. It does however print, in window #0 (the command window at the bottom of your screen), the names of the files that have been copied throughout the procedure. These commands work with any devices you happen to have, be they floppies, microdrives, or ram discs.

There is also a PRINT_USING command. This command will let you pre-define/justify your output. The most typical use for it is numerical justification when I have ng checks. If you wanted to right-justify, and fill the spaces to the left in a numeric field for instance, you would use PRINT_USING O\$***,***.** Then if the numeric data was 1234.56, the resulting field would be, \$**1,234.56 The PRINT_USING command will also work with strings. I haven't used it for this purpose yet but it does make for some interesting possibilities.

QL cont...

For the programmers there are two commands of particular interest. HEX\$ and BIN\$ do pretty much what you would expect them to. They convert to hexadecimal and binary respectively. To convert from hex or binary there are two other commands, HEX and BIN. (How inventive.) Before I got a calculator that did these conversions I had to work them out by hand or write a program to do it. It was a real pain to have to stop writing a program, save it, load the program that did the conversion I needed, do a couple conversions, and then try to get back into my train of thought. A lot of time gets wasted that way!

These are just a few of the wonders that await you when you enter the command TK2_EXT. I have barely scratched the surface of all the timesavers that Toolkit II delivers. One thing I should mention here though, for those of you that get Toolkit II with the Trump Card, the manual you get with it is SO SMALL! Not in the amount of information, but actual physical size. It's 3 3/4" X 5 1/4". Now that's small! If you've got great eyesight you could read it, but don't count on having that great eyesight very long. Anyone that is interested can receive a copy on disc from me (you pay postage). There is also a program included that prints the whole thing out on real sized paper. Send me a microdrive cartridge or 3.5" disc along with a couple bucks and I'll copy it and return them to you.

Next month we could get lucky and win the lottery! Till then, Happy computing and let's keep these machines alive!

From the S.E.T.U.G. SWYM Newsletter SWYM

MORE TASWORD II

Dick Wagner

Some of us have used a program for some time, obtained more/newer equipment, and then find that we have to adapt that old familiar program to use it. This has happened to many of us who have used TASWORD II back in the early tape drive days.

There have been articles and word of mouth explanations on fitting this program to different printer interfaces. In another issue I will guide you thru the process of changing the HELP page to match those changed printer codes, printer name, etc. Why keep a crib sheet of your changes when you can correct the menu?

Here is a review of several popular interfaces. POKE the code numbers into the corresponding addresses, either in direct mode or use POKE program. BUT first, run an address/code check with a loop program to see if there are some that do not need changing. maybe just a few POKES will do it.

AERCO IF		A & J OLIGER/ HACKSEL	
ADDRESS	CODE	CODE	CODE
57999	127	65	127
58000	203	230	203
58001	103	4	103
58005	0	211	0
58006	0	66	0
58007	211	62	211
58008	127	4	127
58009	0	211	0
58010	62	65	62
58011	247	175	247
58013	251	65	251
58014	219	201	219
58015	NC	NC	127

MASTERING "MERGE" (LKDOS) (that is.) from Bob's Notebook.....

As the title implies, this is about exploiting the LKDOS MERGE command to accomplish a special task.

Just recently, I was tinkering with an old BASIC program that I typed in years ago from a book on the ZX81; perhaps you remember it: it was called G.P.G.P (for General Purpose Graph Plotter). I had adapted it to the TS2068 about 1984 and used it for a few years tracking Gas and Hydro usage and later Gas Mileage/Kms for my car. The graphs were of the vertical pillar type as opposed to pie or line graphs. In my adaptation, the only "colouring" was white and black to differentiate between two sets of data on the same screen or printout.

The Larken LKDOS version 3 came along and I decided to use the extended command CIRCLE to fill the pillar graphs with different shadings. This allowed me to display three sets of data on the same screen or printout, which was the original intent of the ZX81 program.

Suffice to say, the BASIC had developed quite a few changes from the earlier version I had been using. But now I had a new BASIC file without the data which had previously been entered; in fact I had a total of nine such files to convert and I did not relish the idea of re-entering all this data from scratch. I cast about for a way to preserve the data (ie, the variables) and transport them to the newly amended BASIC files. Could I MERGE the values into the new BASIC?

I dug out the Larken manual and reread the MERGE part. Here's what it says (in part):
"The MERGE command differs from the cassette MERGE in a few ways. (One of them is that it) doesn't merge program variables, only the program lines. This makes your BASIC overlay programs easier to program."

What this meant was that I could not MERGE the variables into the new BASIC file, but I could MERGE the new BASIC file into the variables. I had never done this before but now was the time to try. Let's call the old file "graph0.Bx" and the new one "graphN.Bx".

The procedure I used was straight-forward enough:

1) LOAD the old program with all its variables intact (RANDOMIZE USR 100: LOAD "graph0.Bx") and;

2) DELETE all the BASIC lines (DELETE 1,9999). Without a listing, the screen was blank but all the variables were still there. This could be demonstrated by a simple PRINT (variable name). At this point, I could have saved the variables under a different BASIC name, but this was not necessary or desirable for this particular exercise.

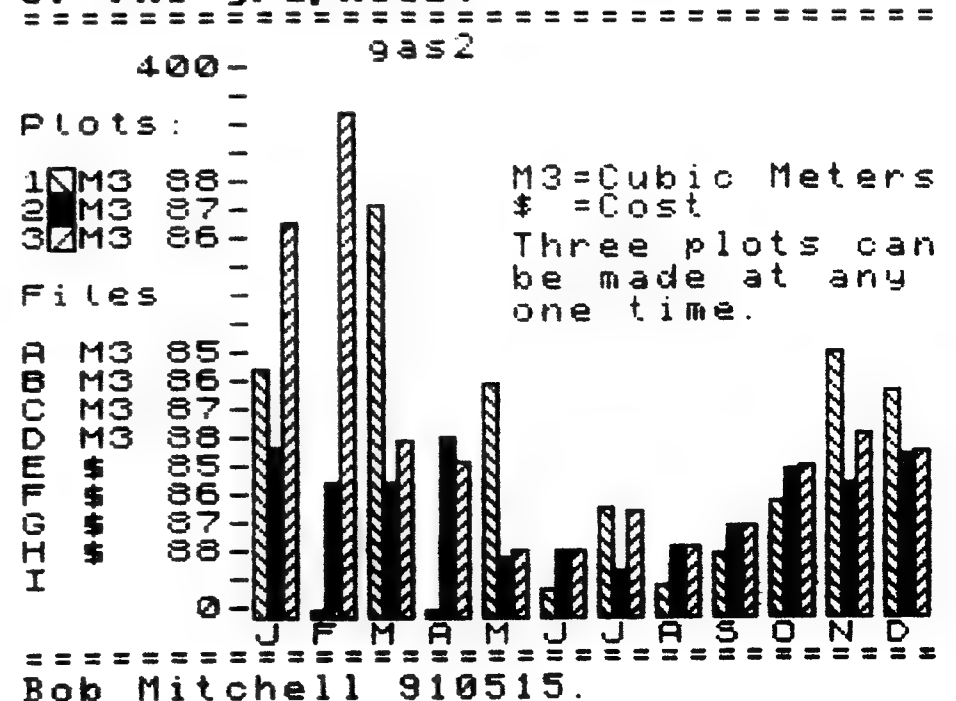
3) Instead, the next step was simply to MERGE the new BASIC (RANDOMIZE USR 100: MERGE "graphN.Bx"). Note that there was no need to save the variables as a CODE file with all the complex programming that entailed. Only BASIC files are used in this method.

It was an odd use of MERGE because there really weren't two files of lines to work on. But the original variables were now incorporated into the new program. Warning! It can take a few moments for the LKDOS MERGE routine to do its work but then the program is there ready to use.

To recap, remember you cannot MERGE variables even in their own separate program. You have to LOAD them first and then MERGE the BASIC listing.

This is not a feature that you might use very often but make a note of this for the future. Once again, the LKDOS V3 was shown to have another gem that confirms it to be the best of the TS2068 DOSs. Can any of the others do as much?

PS. If there are any requests for the GPGP file, I can put a copy into the club library. Here is a sample output from the program to give you some idea of the graphics.




```

605 IF q$=" " THEN PRINT AT d,
11; OVER 1; INVERSE 1;"
"
610 IF q$=" " THEN LET d=d+1
620 IF d=22 OR a$(((c-1)*20)+d-
1)=" " THEN LET d=2
622 IF c=x AND d=22-(c*20)-p TH
EN LET d=2
625 IF q$=" " THEN GO TO 590
630 IF q$="6" AND a$(i)<>"
" THEN LET c=c+1
640 REM IF c>x THEN LET c=x:
BEEP .1,10: GO TO 600
650 IF q$="7" THEN LET c=c-1
660 IF c=0 THEN LET c=1: BEEP
.1,10: GO TO 600
665 IF q$="6" OR q$="7" THEN L
ET d=2
670 IF q$=CHR$ 13 THEN GO TO 6
90
680 GO TO 540
690 LET i$a$(((c*20)-19)+d-2)
695 FOR i=1 TO 10: IF i$(i)=" "
THEN LET i$=i$( TO i-1): GO TO
700
697 NEXT i
700 CLS : PRINT AT 10,8;"Loadin
g... ";i$
705 FOR i=65368 TO 65367+LEN i$
: POKE i,CODE i$(i-65367): NEXT
i: POKE i,CODE "*"
740 REM ! CLOSE #
750 STOP
9000 PRINT USR 100: SAVE "diskme
.B2"

```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

I recently purchased a Larken Disk System for my TS2068 and found myself frustrated by the lack of a simple and short disk file LOAD program where I didn't have to type each file name into a LOAD menu program. I decided to use the SCREEN\$ token to read file names from the screen following a disk CAT. The following brief program will do just that, and can be modified to CAT and LOAD from any disk system by changing the lines annotated below. Carefully type in the following program and change the CAT and LOAD commands in lines 30 and 150 to suit your disk system. I saved the program as an AUTOSTART

file. The program will CAT the current drive and display the contents in the normal way. If the screen is full and you get the SCROLL? prompt, you may either press ENTER to continue the CAT or press the BREAK key to select and LOAD the highlighted file on the screen. ON ERR is used to allow you to LOAD a file found on the screen instead of SCROLLing to the next page. This program is user configurable by allowing you to change screen colors and selection keys. Feel free to add your own enhancements/options.

Happy TIMEXing...

LOADER

By Michael J. Di Rienzo
February 1990

```

10 POKE 23658,8: PAPER 1: BORD
ER 0: INK 9: CLS : DIM C$(10)

```

(^-Sets caps, screen colors and
selects bar width

```

20 ON ERR GO TO 40

```

(^-BREAK brings a select bar.)

```

30 RANDOMIZE USR 100: CAT "",

```

(^-Larken CATalogue command.)

```

40 PRINT #1;AT 0,0;"'ENTER' ad
vances Cursor, Space Bar select
s file, 'X' CATs Disk."

```

(^-User instructions at input line.)

```

50 FOR L=0 TO 21

```

(^-Sets 10 character select bar.)

```

60 LET P=0: LET Q=9: GO SUB 16
0: LET P=16: LET Q=25: GO SUB 16
0

```

(^-Select bar moves in 2 columns
of Larken CAT screen. Bar
alternates between columns.)

```

70 NEXT L: GO TO 50

```

(^-Bar wraps back to line 1.)

```

80 LET B$="": LET E=10: ON ERR
RESET

```


(^-ON ERR GO TO 50 cancelled.)

90 FOR C=P TO Q

(^-Reads highlighted file.)

100 LET A\$=SCREEN\$ (L,C): LET B
\$=B\$+A\$

(^-name from screen.)

110 NEXT C

120 FOR F=LEN B\$ TO 1 STEP -1:
IF CODE B\$(F)=32 THEN LET E=E-1:
NEXT F

(^-Removes any spaces to the right
of the file name.)

130 LET B\$=B\$(TO E)

140 PAPER 7: INK 0: BORDER 7: C
LS : PRINT #1;AT 0,0; FLASH 1;"L
OADING..."; FLASH 0; PAPER 2; IN
K 9;B\$

(^-Prints LOADING file message,
resets ON ERR, allows error
messages.)

150 RANDOMIZE USR 100: LOAD B\$:
STOP

(^-Larken LOAD line.)

160 PAUSE 10: PRINT OVER 1; PAP
ER 2; INK 6;AT L,P;C\$

(^-Prints select bar over file
name. Sets bar speed.)

170 IF INKEY\$="" THEN GO TO 170

180 PRINT OVER 1; PAPER 1; INK
9;AT L,P;C\$

(^-Removes select bar.)

190 IF INKEY\$="X" THEN RUN

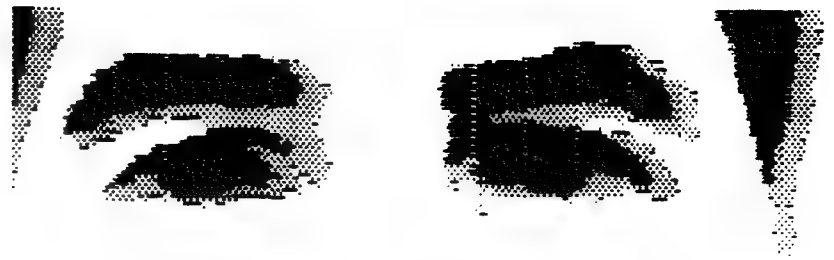
(^-Detects key to re-CAT.)

200 IF INKEY\$=" " THEN GO TO 80

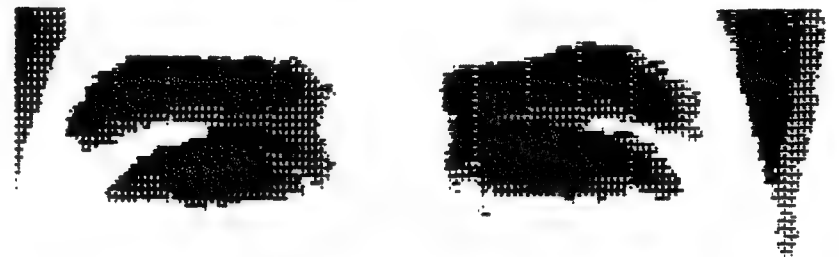
(^-Detects key to select
and LOAD the file.)

210 RETURN

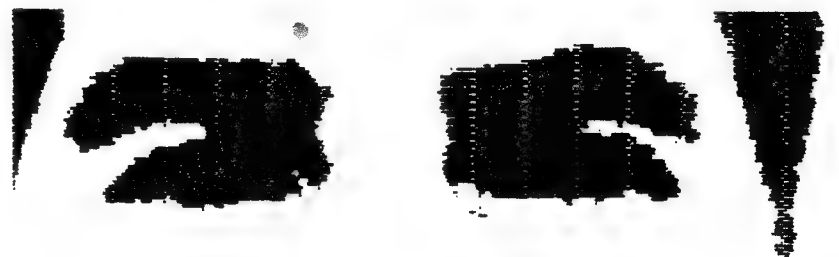
from The Plotter Newsletter



I don't really have a beard, it's just



my five o'clock shadow (at two A.M.)



showing with the light source above my head.

"VIDEOTEX v1.5"

by Jeff Taylor

A problem with the SMUG digitizer is that it has no contrast control. Everything shows up as black or white with no shading. Consequently, in order to display certain details of your subject, you must blast others into deep black oblivion. Each picture you keep is a compromise of trying to show as much detail without making everything a even shade of jet black.

John McMichael has the solution. He has produced a video texturizer (shader) program, *VIDEOTEX*, for the SMUG digitizer. The main feature of *VIDEOTEX* is that it generates a master video file of thirteen screens, each grey-scaled slightly darker than the previous one. This way you can view and print shaded pictures.

The program comes on cassette in two parts, BASIC and CODE. The BASIC section contains all the SAVE and LOAD routines and can be easily modified to suit any disk system. To run it, it is assumed that you have already set up your digitizer per the instructions supplied by SMUG. After you have done this you will never need to use the software supplied with the digitizer again, *VIDEOTEX* is all you need.

Once you have loaded both sections, the menu presents itself. The first thing to do is to set the brightness. Press the brightness adjust key and the program will start showing the first and last scan screens of your capture. If you are using a tv camera, you can vary conditions by changing the camera aperture and/or how much light is on the subject. If you are using a vcr then you may have to alter the brightness setting on the digitizer board itself. Once you are satisfied with the brightness, press any key to return to the main menu.

You are now ready to capture your master video file. Press the "C" key and you are given the option of either running the four-scan capture automatically or with a pause after each scan. Usually you choose the automatic mode. The four scans take about a minute and then you are returned to the main menu. At this point you can either view your captured video file or save it. To view it, simply press "V" and you are then shown one of the thirteen grey-scaled screens. To see the next screen simply press the "5" or "8" key to move through the file. At any time you can print an individual screen on the 2040 printer or save a screen to disk or tape. See a few samples on the preceding page. Pressing "R" returns you to the main menu where you can save the whole file or if you have already saved one, load in a new master file.

Another feature is the ability to overlay one picture on another. this requires some careful manipulation of your screens.

The individual screen saves can also be imported into programs like *Art Studio* where you can further refine the end product before you print them.

Version 1.5 is an upgrade from John's original 1.0 and now includes automatic filtering to remove unwanted individual dark pixels. Video files saved from v1.0 can be imported into this version for "cleaning-up".

For anyone who has invested in the digitizer, this is a "must-have" program. It is simple to operate and the documentation clearly explains each step of the process.

For more information contact John McMichael, 1710 Palmer Drive, Laramie, Wyoming 82070.

May/June 1991

June 7, 1991

Dear OOT Members,

Bob Mitchell and I have been doing some practicing with exchanging programs by modem. While we have both had modems for years, I'm afraid that neither of us have really used them enough to get familiar with their usage. Anyway, we have been sending various files back and forth, and I expect that we will each have an article for the next newsletter. Sort of oriented to the duffers in this subject. Well, that's what mine is going to be like!

What is happening to SNUG? By a reading of various newsletters, not too much. Seems that Don Lambert, who happens to be one of our out-of-town members had offered to become the SNUG newsletter Editor, and the offer was taken up. But after a while he was frustrated by a certain malaise in that organization, which prevented his efforts from getting off the ground.

Well, just recently there has been a suggestion made in the Chicago Area TSUG newsletter that Don just proceed on his own to put out a newsletter for SNUG. It was further suggested that each T/S club nominate a member to assist Don in his efforts. Well, that's where it seems to stand at the moment. I have to confess that I have a bit of doubt about the feasibility of the idea. I rather think that the active Timex owners are already quite involved with their own clubs, and will not have the same enthusiasm to put into another one. Well, maybe I'm too pessimistic. I simply see myself as putting too much time into our own club, let alone consider working another one!

Just today I received a copy of the New England QLUG newsletter which says that they have received the first issue of Don L.'s SNUG newsletter, which is named "ZXir Qlive Alive! Newsletter". And also today I received a letter from Don L. He writes..."April was a fast month for me with all the excitement of getting ZXir Qlive Alive! published from start to finish in one month. I have already started material for the next quarterly issue. But I have had almost no input, but it hasn't been that long since it was mailed. If there is a weak response I don't know what the next move will be."

Bob Mitchell and I have been reworking the OMNIBUS disk in our library. Disk # 2, that is. Essentially we have been trying to make it more flexible for use with systems that have only two drives and no RAMdisk. Well, it works much more smoothly if one has a RAMdisk, but with the modifications it is very easy to modify/customize it for

your own application. I have sent copies out to several members to try it out, shake it down, and come up with any ideas to improve it. Ask for a copy if you are interested. It comes on a single DSDD disk, and is loaded with clever ideas.

Joan Kealy has sent us a series of programs. They are language tutoring programs in German, French, and Spanish. They will be going on another Club disk. Ask for it if interested. I have not assigned a number to it yet.

Joan has also sent two programs containing a host of TS2068 programming tips. I shall have to find a slot in our club library for them also. They are in a Basic program, and Bob Mitchell and I have been trying to figure out how to get them into a Pro/file database. That would make it easier to search out a topic that the Basic program allows. But it is tough sledding, and I'm not at all sure we are going to succeed. In the meantime I have been modifying the INPUT routines, making them operate more smoothly. In the process, learning things about the 2068 that I never know before. Really clever things in it; ask for Joan Kealy's "TIPBITS" programs.

When I looked over the newsletter I came across a couple of things that need some explanation. We'll cover them properly in the next issue but for now I'll mention them so that you aren't left puzzled. In Bob Mitchell's article at the top of page 5 he refers to PPAS. I asked Bob what that meant. He referred me to the Larken Disk system operating manual. Look on page 9 towards the bottom. There you will see that PPAS is the name opposite the address 16093, along with a description.

Also Bob told me that the article has a mistake in the first paragraph following the heading "Explanations:" It reads "...2580 puts the value 16100 into LKDOS address 16100 and....". It should read"...2580 puts the value 16100 into LKDOS address 8214 and.....".

And there's something else about the newsletter that I don't have an answer for, yet. On page 10 we have Part 2 of a series on SuperBasic (for the QL) by Howard Clase. But we have not published a Part 1. This came up at the meeting, and I'll have to ask Howard, or maybe Hugh Howie if they have the answer. Maybe it's me; maybe I misplaced the article! We simply don't know, at this point. We'll explain in the next issue.

You will notice an article by Richard Hurd called Bug Alerts and Updates. The Larken Utilities disk version 2.3 that he mentions, is in our library, as an updated disk #3.

I have a copy of the documentation of the HiSoft C language for the Spectrum, if anyone is interested in looking it over.

We also have a copy of the Logical Disk Management System written by Bob Swoger of the Chicago Area Timex User Group. Bob is anxious to make it available to all Larken users and is offering it at nominal cost. It is available from him, and also RMG Enterprises mention that they will be offering it shortly. Bob has sent me a copy, which I propose to place in our library. Ask me for a copy if you would like to try it.

It may be that our OMNIBUS is more attractive to Larken owners of a RAMdisk, who are capable of customising the OMNIBUS program to their system. The Logical Disk would not require any particular customization; you simply add it to each disk. But you might try a still simpler Disk Management System, one by BytePower called "HELLO", written by Kristian Boisvert, which is on our club library disk #32. Try them all, and take your choice!

We have received some ZX81 Hi-Res programs recently (6 weeks ago) from the Vancouver area. I have not seen them yet; I cannot make copies of them because they require a full 64K memory. Jeff Taylor and Rene Bruneau tried to make tape to tape copies but that did not work out. They are now making up new "originals". We have promised them to Mel Richardson and Philip Joe, and another member whose identity escapes me for the moment. I'll get them out as soon as I get my hands on them.

Is anyone out there still awaiting anything from me. I put a burst of energy on recently and got most everything out. Now it is starting to trickle back, and the routine starts over again!! You can't win! Let me know if I'm not responding; it might be your letter has fallen down the crack somewhere.

Well, that's about it for this letter. I'm sure that after I get this made up I'll remember a host of things I should have said!!

If you have any material for the newsletter do please send it in to me or to the newsletter Editor. That's what keeps our newsletter thriving.

Sincerely,

George Chambers

I've run out of things to say: Wait a minute, I have a couple of programming tips that I have picked up lately.

We know that you can switch from the 2068 mode to the Spectrum mode by using the instruction OUT 244,3. But it has never been easy to get back to the 2068 mode, even when using the command OUT 244,0. It should work, but does not do this reliably. I have been experimenting and I find that this instruction will do it very gracefully. PRINT: OUT 244,0

That will do it. Or try COPY: OUT 244,0. Don't ask me why; I spent part of an evening putting various combinations together and came up with these. Not that you have to do that very often, but I have been asked on occasion for that solution.

Bob Mitchell has found a use for the Larken instruction PRINT #4: DATA 0. Seems this command has the same effect as the NMI-button/F-key operation. It does a call to a user-defined routine that has been placed in the Larken RAM. good for putting into a program line. We used it extensively in the OMNIBUS suite of programs, to get from a program back into the OMNIBUS menu.

And here's one from Joan Kealy's TIPBITS program: For the TS2068.

To LPRINT all lines that are PRINT lines, POKE 26697,80; to restore POKE 26697,83.

Tim Swenson has started putting out a small newsletter dedicated to the QL applications. He has sent me copies of the first three issues. I have written to him asking if he would mind us making copies to send out to our member QL owners. I have not heard from him yet. Just on the possibility that he would not mind. I'll give you his address so you can drop him a line if you would like to subscribe to it. Tim is a member of the CATS group, and a subscriber to our newsletter.

Timothy Swenson
4703 W. Braddock Road
Alexandria, VA 22311

TORONTO TIMEX-SINCLAIR USERS CLUB

May 6, 1991

14 Richome Court,
Scarborough, Ont. M1K 2Y1

Les Cottrell
108 River Heights Drive
Cocoa, FL 32922

Dear Les,

Thank you for the return of the disks. And for looking at the OMNIBUS disk. I enclose a couple of lines from the program which gives a different way of renaming a program. Although this one, I think, does not rename the individual tracks of the program.

I re-did the Starstrike program on the games disk. I was unable to get the Herbert's Dummy Run program to function. I used the "cracker.B1" program on it but to no avail. Seems to have something seriously wrong with it. Almost as though it was not the same program at all, maybe just rubbish. I do not have another copy, so I shall have to keep a look-out. Not that it really matters, I never have played it. It is more a matter of it being part of my collection!

Whether the OMNIBUS AUTOSTART and the three screens should be kept on the DATA disk is a matter of personal preference. If you maintain a back-up of them on another disk there is no reason to keep them on the disk. Primarily, it depends on whether you have enough space on the disk for them. If you have a quad drive, then the space is not so much of a problem. I happen to keep a copy on my quad drive, but it is probably there because of an inadvertent SAVE! But you should keep a copy of these four programs somewhere, once you have personalized them. There's no where else to get them otherwise.

I agree with you about calling the disks differently. Well, I don't know whether "menu" is appropriate, but there really is three types of disks to be identified. There's the one which holds the AUTOSTART and the three menu screens. If it was on RAMdisk then you would put a few other programs on it as well. Then there is what I called the DATA disk; it is really a "Programs" disk, and the third disk would then be the "DATA" disk.

On my system I use the RAMdisk for the four program plus my most often used program such as Tasword, Mscript, Format, Profile, HELLO, and others. Then my Drive 2 holds the "Programs" disk, and Drive 3 holds all my Mscript, Profile, Tasword, and Disk index data files.

I reserve Drive 0 and 1 for my day to day uses. That is where the HELLO program comes in useful, to access the disks in these drives.

You may experience difficulty with some of the programs on the OMNIBUS disk. The problems relate to the use of the LPRINT function. Bob Mitchell uses a FASTEXT 80 printer, and has it on sort of permanently. That's all very well. But when others use the OMNIBUS without having a large printer, or not having it turned on, there's a problem. The programs appear to be locked up. If you use the BREAK key to break out of it you will see that the program is hung up on an LPRINT command that is hung because there is no large printer to accept it. Bob places the LPRINT command to condition his printer.

I think the answer is to put in a conditional command which in effect does a "If big printer is in place and turned on then do an LPRINT". "IF IN 127 = 235 THEN LPRINT" would do the trick on my FASTEXT 80 printer, I think.

The "235" number has to be changes to suit the response of a particular printer. Printers differ, I believe.

Also, several of the screen-copy programs are based on the use of the Fastext 80 printer, and probably will not work with other printers. We have the Basic program that was prepared ready to compile with TIMACHINE. If you want, I can send them to you. Mention the ".Cc" programs for which you would like the Basic.

Glad to hear that Ken Sh. is working on his disassembler program, again. If you send me a copy when it is ready, I can forward a copy to Bob Mitchell. Bob and I have been practicing sending files via modems lately, so we could send this one also.

Shall close this off now and mail it off with the newsletter.

Sincerely,
George Chambers

```
5320>PRINT #od:LOAD "format.B1":REM M
5330 LET u$=" RENAME FILE ": GO SUB d9: INPUT AT oo,oo;"old name
(in full) ": LINE o$!AT oa,oo;"new name(in full) ": LINE r$: PRI
N #od: MOVE o$,r$: PRINT #od: DATA 0: REM N
5340 PRINT #od: LOAD "lk2cy.B1": REM O
5350 GO TO m1: REM P
5360 GO TO VAL "4070": REM O
5370 GO SUB VAL "2610": GO TO m: REM R
5380 BORDER og: PAPER og: INK oo: CLS : LET help=oo: GO SUB VAL
"1030": GO SUB VAL "2390": INPUT "l=more 0=menu ":mm: IF mm THEN
```

```
5320>PRINT #od:LOAD "format.B1":REM M
5330 LET u$=" RENAME FILE ": GO SUB d9:
INPUT AT oo,oo;"old name(in full) ": LINE
o$!AT oa,oo;"new name(in full) ": LINE
r$: PRINT #od: MOVE o$,r$: PRINT #od: D
```

